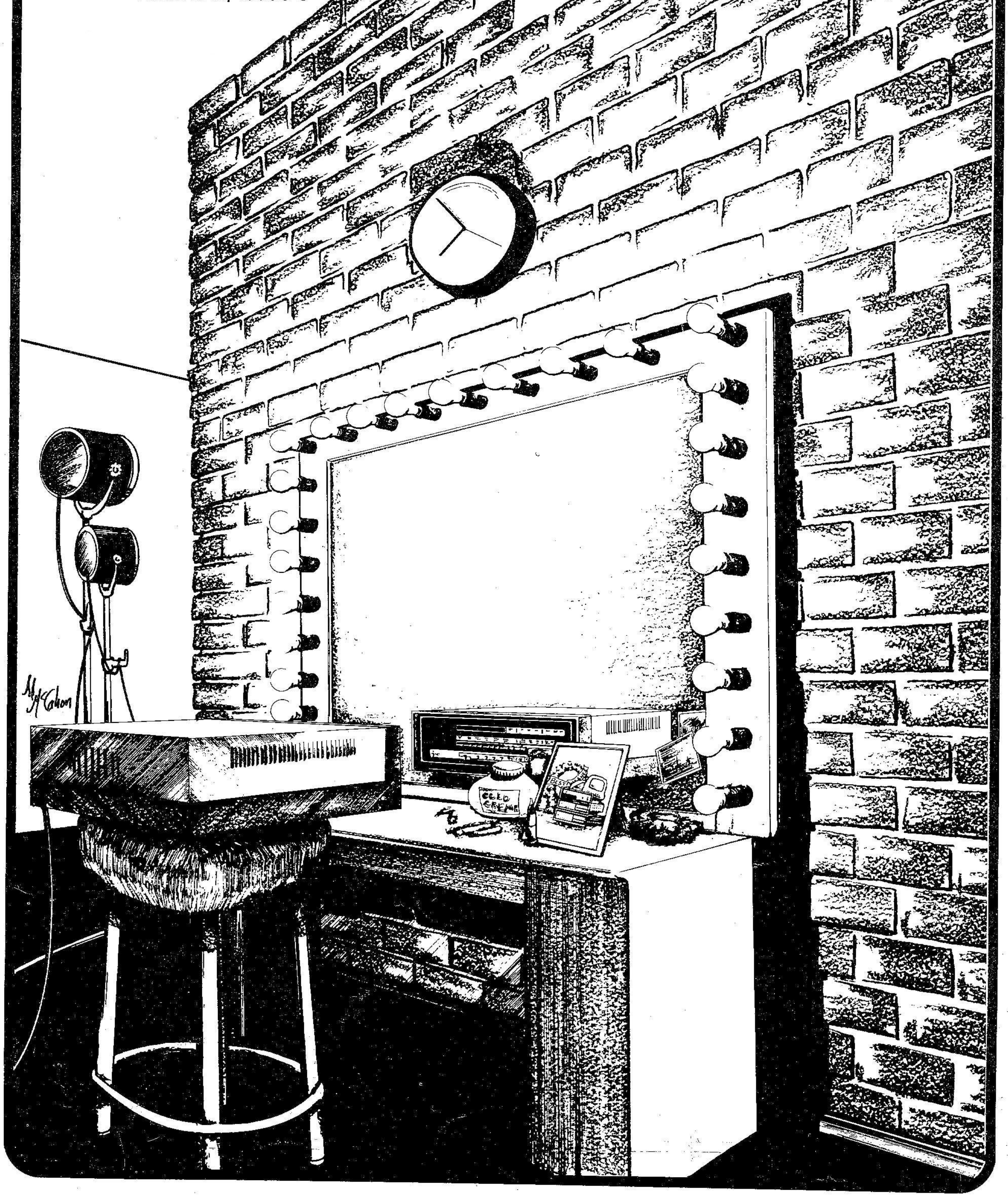
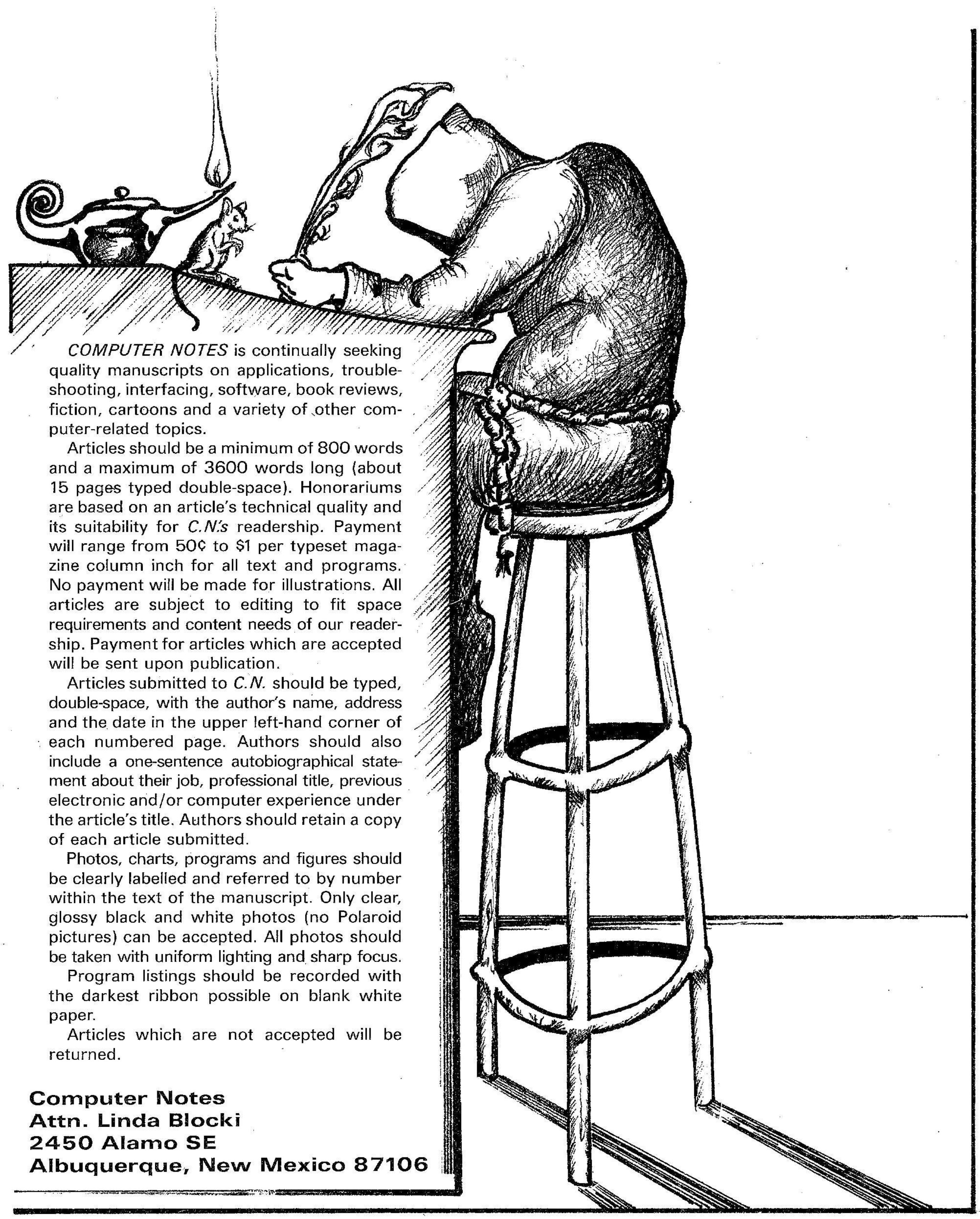
computer notes

March '77

50¢

Volume 2, Issue 9





COMPUTER NOTES is continually seeking quality manuscripts on applications, troubleshooting, interfacing, software, book reviews, fiction, cartoons and a variety of other computer-related topics.

Articles should be a minimum of 800 words and a maximum of 3600 words long (about 15 pages typed double-space). Honorariums are based on an article's technical quality and its suitability for *C.N.'s* readership. Payment will range from 50¢ to $1 per typeset magazine column inch for all text and programs. No payment will be made for illustrations. All articles are subject to editing to fit space requirements and content needs of our readership. Payment for articles which are accepted will be sent upon publication.

Articles submitted to *C.N.* should be typed, double-space, with the author's name, address and the date in the upper left-hand corner of each numbered page. Authors should also include a one-sentence autobiographical statement about their job, professional title, previous electronic and/or computer experience under the article's title. Authors should retain a copy of each article submitted.

Photos, charts, programs and figures should be clearly labelled and referred to by number within the text of the manuscript. Only clear, glossy black and white photos (no Polaroid pictures) can be accepted. All photos should be taken with uniform lighting and sharp focus.

Program listings should be recorded with the darkest ribbon possible on blank white paper.

Articles which are not accepted will be returned.

Computer Notes
Attn. Linda Blocki
2450 Alamo SE
Albuquerque, New Mexico 87106

COMPUTER NOTES is published monthly by MITS, Inc., 2450 Alamo SE, Albuquerque, NM, 87106, (505) 243-7821. A free year's subscription is included with every purchase of an Altair™ computer. Regular subscriptions can be ordered from the MITS Customer Service Dept. for $5 per year in the U.S. and $20 per year for overseas. Single copies are available for 50¢ each at all Altair Computer Centers. Send articles, questions, comments and suggestions to Editor, *COMPUTER NOTES*, MITS, Inc.

Computer Goes to Court

Dear Ed.,

This has got to be one of the most bleak and bitter winters in the Big Apple. It is a chore to get to the neighborhood library (on staggered hours because of severe budget cuts) to have the article from the New York Post xeroxed, so I typed it out for you.

I am always on the alert for the imaginative and humane use of the computer and was fascinated by this particular application.

Like a great many other citizens--concerned, interested and affected--I would like to see some of the results of this experiment. Is there any way you could follow up?

Sincerely,
Sydell Rosenberg
141-30 Pershing Crescent
Apt. 3G
Jamaica, NY 11435

New York Post: October 7, 1976

New Friend of Court: A Computer

A computer link between the city's building code enforcement unit and the Housing Court is being installed today.

For the first time a hearing officer will be able to press a few buttons and receive on a TV screen an instant report of the number of violations in an apartment building and other pertinent data. Until now, such information has not been immediately available and a hearing officer would be forced to rely on testimony from a tenant and landlord.

Early this week while a machine was being tested in a courtroom, a landlord testified that he operated a clean building. The hearing officer, anxious to see the machine work, asked the mechanic for a demonstration on the building in question. When the picture came into focus, it showed the building had 41 violations.

"I didn't say it was perfect," the embarrassed landlord explained. Mayor Beame and housing officials are expected to attend installation ceremonies today at the court, 111 Center Street.

The computer hookup, urged by the Community Service Society, a non-sectarian social agency, will be used by five hearing officers in Manhattan for a six-month trial period at a cost of $14,000 donated by the New York Community Trust. According to Bruce Gould of CSS, the computerized data will reduce trial time, aid in settling penalties, and provide a tool for recouping money from landlords for emergency repairs made by the city. The CSS housing and urban development unit also succeeded in having a state law enacted which makes the computer admissable as prima facie evidence, reducing the need for subpoenaing official records.

This story is being investigated further. Additional details will be published in C.N.

Texas Conventions Slated

The 15th ANNUAL CONVENTION OF THE ASSOCIATION FOR EDUCATIONAL DATA SYSTEMS (AEDS) is scheduled for April 25-29 this year at the Green Oaks Inn in Fort Worth, Texas.

For information about exhibits, activities and accomodations, contact:

Alton R. Goddard
Publicity Chairman
AEDS
1201 Sixteenth St., NW
Washington, DC 20036
(202) 883-4100

The 1977 COMPUTER USERS CONFERENCE will be held March 25 at East Texas State University, Commerce, Texas.

The conference will focus upon both large and mini/micro systems.

Industrial and educational representatives will conduct panel discussions on computer usage trends and needs in their respective areas.

There will be a $20 charge ($10 for students) for the conference. For further information, contact:

Donna Hutcheson
Computer Users Conference Coordinator
East Texas State University
Department of Computer Science
Commerce, Texas 75428
(214) 468-2954

Editor Andrea Lewis
Asst. Editor Linda Blocki
Production Tom Antreasian, Al McCahon, Steve Wedeen, Grace Brown
Contributors Dave Antreasian, Tom Durston, Bruce Fowler, Rich Haber, Randy Huddleston, Mike Hunter, Sondra Koppenheffer, Bill Kuhn, David Le Jeune, John Poe, Rick Ranger, Bob Scott

mits
2450 Alamo S.E.
Albuquerque, New Mexico 87106

IN THIS ISSUE

Altair 88-ADC useful for Digitizing Waveform

By Dave Antreasian, Rick Ranger and John Pope

One of the most useful applications for an 88-Analog-to-Digital Converter (88-ADC card) is that of "digitizing" an analog waveform.

Let's look at the example of a time-analysis of stress placed across a lever used in some type of mechanical application. (The following discussion can just as easily be applied to a study of a microphone's low-frequency response, time-analysis of a camera shutter or any other time response problem.)

A strain gauge, which is a device that outputs a voltage proportional to the stress applied across it, must be used. Suppose that the output of a sensor looks like the curve in Figure 1.

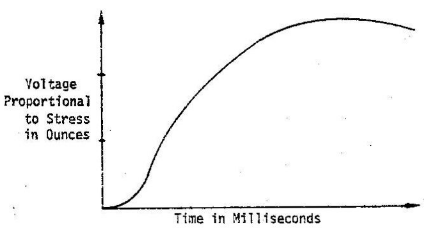


Figure 1

To store data that represents this curve, we need a device which can quickly sample the analog level at a number of points along the curve and then convert the levels to digital numbers that the computer can process. In this way the actual curve can be reconstructed or simulated with a digital "staircase" curve. (See Figure 2.) Note that the more samples taken within a given period, the more accurate the reconstructed curve. In very fast response curves, the curve variations enlarge between the sampling points. This makes the reconstructed curve discontinuous or misrepresentative of the original waveform, if the converter cannot track fast enough. Exactly where this limitation occurs varies with the specific response curve and the type of processing required.

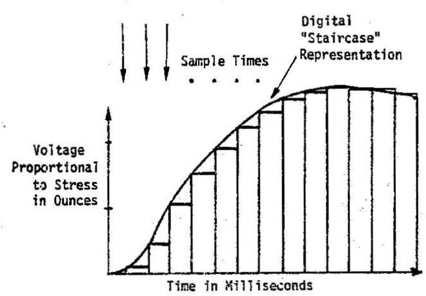


Figure 2

Due to speed limitations in many applications, the only way to perform analysis is to store a sample block of data, analyze or plot the data "off-line," then return and collect another sample of data. "Real-time" processing is realized only with slowly varying signals. However, if no data storage is required, a fast CRT terminal can provide a simulated real-time plot of data.

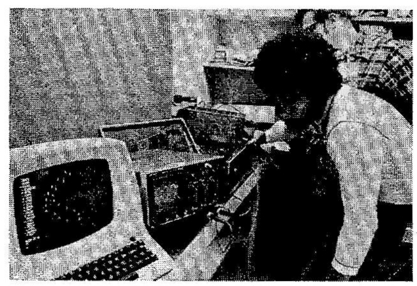
The following two-part program, written in both machine language and BASIC 3.2, combines an easy method of "off-line" data analysis with a high speed data acquisition. The program also includes the "USR" BASIC function. It requires 16K of memory and writes a block of data approximately 8K bytes long into memory. This represents about 4000 samples, since the ADC generates a two-byte word, and corresponds to a data-sampling period of approximately .8 second.

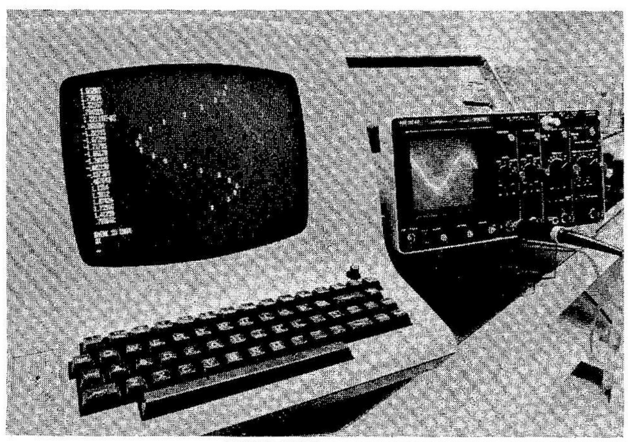
Data sampling time can be extended by allocating more memory space.

After the data block has been written, the machine does a plot of the data or sequentially lists all 4000 sample values on the terminal for visual analysis. The program is written to access multiple channels from a multiplexer, allowing the program to compare the phases of different signals. The number of channels may also be changed from 1 to 96.

Before running this program, which begins on the following page, make sure to limit memory size to 8000.

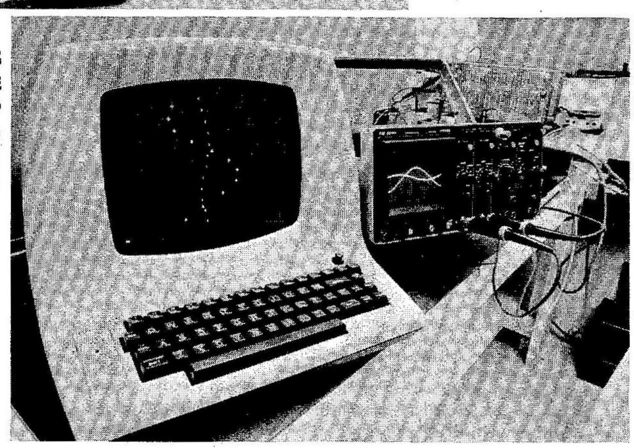
Continued





550 HZ SIGNAL PLOTTED ON A CRT

CRT PLOT SHOWING TIME-PHASE RELATIONSHIP OF TWO ANALOG SIGNALS



Continued

```
THIS PROGRAM IS INITIALLY SET UP TO INPUT FROM ONE OF EIGHT
CHAN.FROM THE EIGHT CHAN MUX AND STORE THE DIGITIZED DATA
STARTING AT LOCATION 037225Q.  THE CHAN.  IS INCREMENTED AND
THE STORAGE LOCATION IS INCREMENTED THEN THE NEW DATA IS
STORED.  THE MOST SIGNIFICANT DATA BYTE IS STORED FIRST
THEN THE LEAST SIGNIFICANT DATA BYTE.  THE MACHINE LANGUAGE
PROGRAM IS FIRST PUT INTO MEMORY AND WILL BE ACCESSED
BY THE USR FUNCTION.  THE MACHINE LANGUAGE PROGRAM MAY
BE EXITED FROM BEFORE THE END OF MEMORY IS ENCOUNTERED
BY TRIGGERING OFF THE STATUS BIT.  THIS MASKING IS DONE AT
LOCATION 037170Q.  THE AMOUNT OF MEMORY TO BE USED AS A
SCRATCH PAD FOR STORAGE MAY BE CHANGED AT LOCATION 037163Q.
THIS LOCATION NOW CONTAINS THE LAST LOCATION AVAILABLE
FOR DATA STORAGE.  A DELAY IS NECESSARY WITHIN THE PROGRAM
TO ALLOW THE A/D TO SET UP AND STABALIZE.  THIS DELAY IS
LOCATED AT LOCATION 037155Q AND MUST BE A MINIMUM VALUE OF
001Q.  THIS MAY BE CHANGED TO GIVE A GREATER DELAY BETWEEN
SAMPLES.  THE FIRST CHAN.# TO BE READ IS LOCATED AT 037141Q
AND MAY BE CHANGED TO START WITH ANY OF 23 CHANNELS.  THE
LAST CHAN.# TO BE READ IS LOCATED AT 037216Q AND REPRESENTS
THE LAST CHAN.#+1 TO BE READ.
```

Run this portion of the program to set up the machine language code starting at location 8000 decimal. (limit memory size to 8000 when initializing basic) After running, type "NEW" and run the second portion of the program (starting with line #1).

```
10 POKE 8000 , 0 :POKE 8001 , 1 :POKE 8002 , 2
20 POKE 8003 , 3 :POKE 8004 , 4 :POKE 8005 , 5
30 POKE 8006 , 6 :POKE 8007 , 7 :POKE 8008 , 8
40 POKE 8009 , 9 :POKE 8010 , 10 :POKE 8011 , 11
50 POKE 8012 , 12 :POKE 8013 , 13 :POKE 8014 , 14
60 POKE 8015 , 15 :POKE 8016 , 16 :POKE 8017 , 17
70 POKE 8018 , 18 :POKE 8019 , 19 :POKE 8020 , 20
80 POKE 8021 , 21 :POKE 8022 , 22 :POKE 8023 , 23
90 POKE 8024 , 24 :POKE 8025 , 245 :POKE 8026 , 197
100 POKE 8027 , 213 :POKE 8028 , 229 :POKE 8029 , 1
110 POKE 8030 , 149 :POKE 8031 , 31 :POKE 8032 , 17
120 POKE 8033 , 64 :POKE 8034 , 31 :POKE 8035 , 26
130 POKE 8036 , 211 :POKE 8037 , 131 :POKE 8038 , 211
140 POKE 8039 , 129 :POKE 8040 , 33 :POKE 8041 , 0
150 POKE 8042 , 0 :POKE 8043 , 35 :POKE 8044 , 62
160 POKE 8045 , 1 :POKE 8045 , 189 :POKE 8047 , 194
170 POKE 8048 , 107 :POKE 8049 , 31 :POKE 8050 , 62
180 POKE 8041 , 64 :POKE 8052 , 184 :POKE 8053 , 202
190 POKE 8054 , 126 :POKE 8055 , 31 :POKE 8056 , 219
200 POKE 8057 , 0 :POKE 8058 , 15 :POKE 8059 , 218
210 POKE 8060 , 131 :POKE 8061 , 31 :POKE 8062 , 225
220 POKE 8063 , 209 :POKE 8064 , 193 :POKE 8065 , 241
230 POKE 8066 , 201 :POKE 8067 , 219 :POKE 8068 , 133
240 POKE 8069 , 2 :POKE 8070 , 3 :POKE 8071 , 219
250 POKE 8072 , 135 :POKE 8073 , 2 :POKE 8074 , 3
260 POKE 8075 , 19 :POKE 8076 , 26 :POKE 8077 , 254
270 POKE 8078 , 8 :POKE 8079 , 202 :POKE 8080 , 96
280 POKE 8081 , 31 :POKE 8082 , 195 :POKE 8083 , 99
290 POKE 8084 , 31 :POKE 8085 , 0 :POKE 8086 , 15
RUN
NEW

1 OUT 130,0:OUT131,255:OUT130,046:OUT128,0:OUT129,255
2 OUT128,046:0:OUT134,0:OUT135,0:OUT134,022:OUT132,0
3 OUT133,0:OUT132,22
4 REM HIT A KEY WHEN EVER YOU ARE READY TO SAMPLE
5 REM AFTER 1 PRINT START
6 REM USR LOC SETUP
7 POKE73,89:POKE74,31
9 INPUT "1=PLOT, 0=NO PLOT";PL
10 PRINT "START":WAIT0,1,1  **
31 S=USR(Y)
35 FOR I=8085 TO 16383 STEP 2
40 X=PEEK(I):Y=PEEK(I+1)
50 V=((16*X+((Y/16)AND 15))*10/4095)-5
56 GOSUB 2000
60 NEXT I
```

Continued

Software Contest Postponed

As we announced in the December issue of Computer Notes, the Altair Users Group Software Library has been moved to Atlanta and will now be handled by the Altair Software Distribution Company (ASDC). During the move, the monthly software contest has been temporarily suspended until early March. At that time all the software entries for December, January and February will be judged, and multiple prizes will be awarded. Winners will be announced in the April issue of Computer Notes.

The ASDC is organizing the Altair User Group Software Library so that each Altair computer center around the world will have the software available in machine readable form with more complete documentation. Until then, however, all software orders from the Users Group Library should be submitted to the ASDC's headquarters in Atlanta, 3330 Peachtree Rd., Suite 343, Atlanta, GA, 30326.

The following programs have been submitted since the last software contest in November and are currently available from the ASDC. These programs will be included with all other entries for the March Software Contest.

12-13-761--$2.00
Author: Willard I. Nico
Length: 9 lines BASIC
Title: "Dec-Hex/Hex-Dec"

Two subroutines: one for converting decimal values to a Hexadecimal string and a second to convert a Hexadecimal string to a decimal value.

12-14-761--$2.00
Author: Frank R. McCoy
Length: 150 lines BASIC
Title: "Basic Renumbering Program"

Renumber and reformat program for programs written in MITS Extended Disk BASIC.

1-18-771--$2.00
Author: Alan R. Miller
Length: 1K
Title: PROMON2

PROMON2 is an improved version of PROMON1K.

1-18-773--$5.00
Author: Darrell J. Van Buer
Length: 1776 lines Assembly
Title: "Multiprogramming with a Variable Number of Tasks (MVT)"

Continued

Software Contest Postponed Continued

Interrupt processing and syncronization for multiprogramming, also provides routines for storage management, for dynamic program control and for dynamic creation and destruction of independently running programs.

1-18-774--$2.00
Author: John R. Lynch
Length: 119 lines BASIC
Title: "T-Twelve-Tone Row Generator"

Output of this program is used in music composition.

1-18-775--$5.00
Author: Henry Everett Lacy
Length: 635 bytes
Title: "Function Package"

Provides functions for decimal support Package #10-15-761. Includes Logarithmic, Exponential, Trigonometric, Real Powers, and has useful conversion routines.

1-21-771--$2.00
Author: Alan Miller
Length: 1/2K + 1K Work Space
Title: "Game of Life"

Game deals with the life in various cells on a rectangular grid.

Altair Computer Courses Offered

BY: Bob Scott
Director of Service at MITS

Beginning this summer, Albuquerque's Technical Vocational Institute (TVI) and North American Technical Institute (NATI) will each offer a computer technology course designed around the Altair computer system.

Each course, based on the popular Altair 8800b computer, will provide technicians and hobbyists as well as the novice computer enthusiast with the valuable opportunity to learn about theory of operation and troubleshooting of the Altair microcomputer.

Roy Stone, director of NATI, said that the course can be applied to an Associate's degree at the Institute.

Walter Rice, coordinator of TVI's Electronics Department, said that the course will be offered at TVI as an optional night class with a pre-requisite of digital electronics.

Based upon student response, both Rice and Stone said additional systems will be purchased for each school, and other courses on computer applications and software will be offered in future semesters.

Look for articles on both of these Altair Computer courses in upcoming issues of Computer Notes.

DIGITIZING WAVEFORM
Continued

```
70 GOTO 10
2000 D=36
2010 A=INT (V*10)
2020 X=D+A
2025 PRINTV;
2026 IF PL=0 GOTO 2040
2030 PRINTTAB (X)"*"
2040 RETURN
RUN

**For 2SIO change to:  10 PRINT "START":WAIT 16,1; (18,1) FOR SECOND
                       PORT
```

```
                    MACHINE LANGUAGE A/D SAMPLE
BASIC WILL RESIDE FROM 0 TO 7999
THIS PROGRAM WILL RESIDE FROM 8000 TO 8084 AND WILL STORE SAMPLES
STARTING AT 8085 AND ENDING AT A SPECIFIED LOCATION.

LOCATION      CONTENTS    MNEMONICS         COMMENTS
-----------------------------------------------------------------------
037 100         000                      ; DATA FILE START
    101         001
    102         002
    103         003
    104         004
    105         005
    106         006
    107         007
    110         010
    111         011
    112         012
    113         013
    114         014
    115         015
    116         016
    117         017
    120         020
    121         021
    122         022
    123         023
    124         024
    125         025
    126         026
    127         027
    130         030                      ;DATA FILE END
    131         365        PUSH PSW      ; SAVE ALL REGISTERS
    132         305        PUSH B
    133         325        PUSH D
    134         345        PUSH H
    135         001        LXIB          ; LOAD FIRST STORAGE ADDRESS
    136         225                      ; START STORAGE AT DECIMAL
    137         037                      ; 8085.
    140         021        LXID          ; LOAD FIRST CHAN OF MUX
    141         100                      ; CONTAINED IN DATA ABOVE
    142         037
    143         032        LDAXD         ; OUT CH.# TO PIA
    144         323        OUT
    145         203
    146         323        OUT
    147         201
    150         041        LXIH          ; START DELAY COUNTER
    151         000
    152         000
    153         043        INXH          ; INCR THE HL COUNT
    154         076        MVIA
    155         001                      ; DELAY TIME (MINIMUM 001)
    156         275        CMPL
    157         302        JNZ           ; IF NOT ZERO GO AGAIN
    160         153
    161         037
    162         076        MVIA          ; CHECK FOR END OF STORAGE
    156         275        CMPL          ; IF NOT ZERO GO AGAIN
    157         302        JNZ
    160         153
    161         037
    162         076        MVIA          ; CHECK FOR END OF STORAGE
    163         100[1]
```

Continued

DIGITIZING WAVEFORM
Continued

```
164    270         CMPB
165    312         JZ          ; IF AT END GET OUT
166    176
167    037
170    333         INP         ; INPUT STATUS FOR AN
171    000 (020)[2]            ; EXIT BEFORE COMPLETE.
172    017         RRC
173    332         JC          ; NO STATUS CHANGE JUMP OVER
174    203                     ; THE EXIT GATE AND CONTINUE.
175    037
176    341         POP H       ; EXIT GATE START
177    321         POP D
200    301         POP B
201    361         POP PSW
202    311         RET         ; END OF GATE, GET OUT
203    333         INP         ; INPUT MSB
204    205
205    002         STAXB       ; STORE MSB
206    003         INXB        ; INCR STORAGE LOCATION
207    333         INP         ; INPUT LSB
210    207
211    002         STAXB       ; STORE LSB
212    003         INXB        ; INCR STORAGE LOCATION
213    023         INXD        ; INCR CH.# LOCATION
214    032         LDAXD       ; LOAD CH.# LOCATION
215    376         CPI
216    010[3]                  ; LAST CH.#+L
217    312         JZ          ; IF ON LAST CH.# JUMP TO
220    140                     ; START CH.#.
221    037
222    303         JMP         ; NOT ON LAST CH.# JUMP TO
223    143                     ; INCR. CH.# AND CONTINUE.
224    037
225    ------------START OF DATA MSB AND LSB------------------------
```

1 Change for additional memory
2 2SIO
3 Change to number of desired channels + 1.

Personality Traits Affect Computer Programming

By Mike Hunter

Most people think that computer programming involves simply learning about hardware and software. However, in THE PSYCHOLOGY OF COMPUTER PROGRAMMING, author Gerald M. Weinberg convincingly presents his theory that computer programming is both an individual and a social activity which is greatly affected by the programmer's personality, motivation, training and the surrounding environment.

Based upon Weinberg's 10 years of teaching at The State University of New York at Binghamton and research at the IBM System Research Institutes in New York and Geneva, the 288-page hardbound book provides a unique insight into many concepts not usually discussed in most programming books or classes. One such concept that he discusses in chapter 10 is how programming can be improved by increasing a programmer's motivation with a higher salary, the opportunity to help plan tasks and more time to complete a task.

Weinberg says in chapter 8 that the various aptitude tests, designed for the purpose of selecting programmers, are for the most part a failure. He says that personality traits definitely make the difference between a successful or an unsuccessful programmer and thus should be considered when selecting a programmer. However, attempts to place each person in the one job best suited for his personality often fail for many reasons. In order to relieve the pressure that forces people to continually adapt to a single job, Weinberg suggests that managers frequently change work assignments and thus encourage egoless programming.

Weinberg, currently Professor of Computer Systems at the School of Advanced Technology at State University of New York, developed THE PSYCHOLOGY OF COMPUTER PROGRAMMING in conjunction with his course by the same title. The book

Continued

Altair Software Features New Look

By Thomas Durston

As the new versions of Altair software on cassette tape become available, users will notice a change in the packaging of the tapes. Each type of Altair software has its own unique color for its plastic case and label, as shown below.

Altair 4K BASIC Version 4.0
Cassette Color - Red
Label Color - Blue

Altair 8K BASIC Version 4.0
Cassette Color - Blue
Label Color - Yellow

Altair Extended BASIC Version 4.0
Cassette Color - Black
Label Color - White

Altair, Package II Version 3.01
Cassette Color - White
Label Color - Yellow

As a special feature, the software has been recorded on both sides of the cassette, giving users an additional copy at no extra cost.

All of these manufacturing changes were made possible by a new duplicating method similar to mass production techniques used for prerecorded audio tapes. This method allows increased production of tapes without sacrificing recording quality.

The recording format has been changed only slightly to accommodate the new software. On the three types of Altair BASIC, the leader byte has been changed to 302 (Octal), and the test pattern (125 Octal) has been moved to after the end of the Altair software data.

88-PCI Article Corrected

In the article, "88-PCI Offers Unlimited Potential" (see pp. 19-21, 23 and 24 of Jan./Feb. C.N.), the propagation delay under relay outputs for the 680b-PCI (p. 23) was mistakenly labelled nsec. but instead should read:

Pull In	3.5 msec.
Release	4 msec.
Bounce	1.2 msec.

ASDC, AUG Functions Outlined

Both the ASDC and the Altair Users Group Software Library are MITS-owned and designed to distribute software for the Altair family of computers. But each organization has distinct purposes and functions. The followint table should clarify any confusion about the ASDC and AUG.

QUESTION	ASDC	AUG
1. How do the organizations differ?	ASDC solicits software systems from professionals, thoroughly tests and evaluates this software, and markets and distributes through the Altair dealer network. The ASDC is profit-oriented both for the author of the software and for the dealer. ASDC makes certain warranties on the software that it distributes.	AUG solicits useful programs and subroutines from all the interested Altair users. This software undergoes little or no testing and is distributed directly to anyone for a nominal copying and handling charge. AUG warrants none of its software.
2. What does it take to submit software?	Individuals must complete the ASDC Software Submittal Packet, available from the ASDC and from local ASDC dealers. The packet is free, but there is a $25 submittal fee to discourage low-quality work and to help offset preliminary evaluation costs.	Submissions must include software narrative, flow chart, code listings, sample run and users instructions. There is a software release form but no formal submittal packet and no submittal fee.
3. How is the software evaluated?	Commercial appeal for this software is determined and then extensively evaluated and tested, with checks made for thorough documentation and error-free code.	A brief examination is made to see if the software is worthwhile and appears to be adequately documented for distribution.
4. How does a user obtain the software?	ASDC software is available only through ASDC dealers throughout the world under the terms of a Limited Use License Agreement. The agreement requires the one time payment of a commercially competitive license fee. All title and other proprietary rights to the software remain in ASDC.	Currently, copies of the AUG Library may be obtained directly through the ASDC office in Atlanta. Future plans are to set up distribution through the Altair dealers.
5. If the software is accepted, what does the author receive?	The author is paid on a royalty basis for each software package distributed through an Altair dealer. Exact royalty percentage is pre-set by the ASDC and the author, prior to final acceptance.	The author receives free coupons to be applied toward other software in the AUG Library. A contest is also held, and prizes are awarded for the best entries.
6. What kind of support does the software get when sold?	All ASDC software is distributed with certain warranties. Software maintenance and special customization requests are handled either by ASDC or the author, according to prior agreement. The dealer distributing the software is responsible for software installation and customer training.	No support or performance guarantees are given with any software distributed through the AUG. Software is delivered to the purchaser "as is."

Continued on Page Nine

Personality Traits Affect Computer Programming

Continued

is arranged like a textbook with a summary and an extensive bibliography as well as thought questions for both programmers and managers after each chapter.

Weinberg wrote the book in an easy-to-understand, nontechnical style in order to encourage the greatest number of people--not just programmers but programming managers and the many other people involved with programming--to read it.

The style is also refreshingly unpretentious. As Weinberg states in the preface, many of the views in the book are merely his own opinions, based on personal observations. Although he admits that some may be "wrong," he suggests that his ideas are not sterile.

Weinberg has begun an intriguing new field of study with THE PSYCHOLOGY OF COMPUTER PROGRAMMING. His book should generate many interesting ideas that will be explored for years to come.

"HIT ME AGAIN!"
Play Blackjack with a Computer

This comprehensive Blackjack program plays the popular poker game according to most of the standard Las Vegas rules. The dealer has to draw to 16 and stand on 17. He also has to offer insurance if his up-card is an Ace. Betting options include splitting a pair and doubling under (doubling the bet for one more card.)

The dealer is generally trustworthy, but if you think he's cheating, you can always hit RESET.

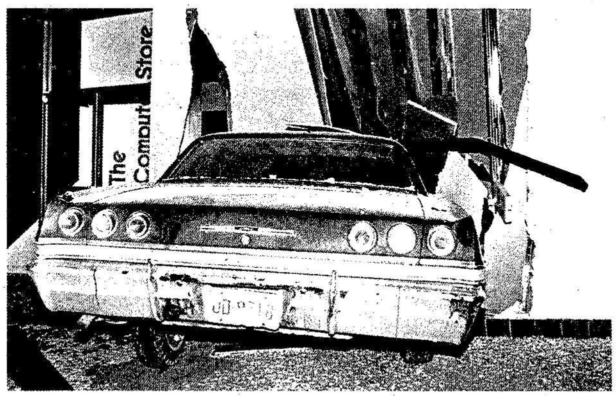
```
1 'ALTAIR LIBRARY #630751
2 'AUTHOR JIM BABCOCK
3 PRINT"LET'S PLAY BLACKJACK!"
4 PRINT"RESPONSES TO 'HIT' :"
5 PRINT"Ø = NO HIT"
6 PRINT"1 = HIT"
7 PRINT"2 = DOUBLE BET & HIT"
8 PRINT"    ONCE MORE ONLY"
9 PRINT"3 = SPLIT THE PAIR"
10 PRINT"BET Ø TO EXIT"
11 X=Ø:W1=Ø
12 INPUT "ENTER A RANDOM # ";N
13 FOR I=1 TO N/2
14 X=INT(1Ø*RND(1)):NEXT I
15 DIM D(52)
16 FOR A=1 TO 52
17 D(A)=Ø:NEXT A
18 DIMQ(52)
19 FOR A=Ø TO 39 STEP 13
20 FOR C=1 TO 13:Q(A+C)=C
21 NEXT C:NEXT A
22 '+++++MAIN PROGRAM+++++
23 K=K+1
24 FOR P=1 TO 5
25 E(P)=Ø:V(P)=Ø:T(P)=Ø
26 NEXT P:V(3)=1
27 PRINT:P=1
28 INPUT"YOUR BET";W
29 W(2)=W
30 IF W<=Ø THEN 289
31 IF W<=100000! THEN35
32 PRINT"THATS TOO MUCH, THE";
33 PRINT" LIMIT IS $100,000"
34 GOTO 28
35 PRINT"I SHOW      ",
36 GOSUB 201
37 IF E(1)=Ø THEN 39
38 V(4)=1
39 V(5)=1
40 GOSUB 201:REM DEAL
41 M=X:P=2
42 PRINT"FIRST CARD IS ";
43 GOSUB 201:REM DEAL
44 G=X
45 PRINT"NEXT CARD IS ";
46 GOSUB 201:REM DEAL
47 IF V(2) >Ø THEN 76
48 S=X
49 IF V(3)<>1 THEN76
50 IF T(P)<>21 THEN 57
51 PRINT"***BLACKJACK***"
52 PRINT"MY HOLE CARD WAS ",
53 X=M
54 GOSUB 229:REM PRINT CARD
55 W1=W1+1.5*W
56 GOTO 270
57 IF V(4)=Ø THEN 70
58 PRINT"INSURANCE ANYONE? ";
59 PRINT"TYPE 1 IF YES  ";
60 INPUT I
61 IF I<>1 THEN 70
62 IF T(1)<>21 THEN 67
63 W1=W1+W
64 PRINT"YOU WON $";W/2;"ON";
65 PRINT" YOUR INSURANCE BET"
66 GOTO 70
67 W1=W1-W/2
68 PRINT"YOU LOST $";W/2;"ON";
69 PRINT" YOUR INSURANCE BET"
70 IF T(1)<>21 THEN 76
71 PRINT"**I HAVE BLACKJACK**"
72 PRINT"MY HOLE CARD IS ",
73 X=M
74 GOSUB 229:REM PRINT CARD
75 GOTO 268:REM PAY UP
76 IF T(P)<=21 THEN 84
77 IF E(P)>Ø THEN 82
78 PRINT"YOUR BUSTED,  ";
79 PRINT"YOUR TOTAL IS ";T(P)
80 IF V(2)=1 THEN 107
81 GOTO 249:REM DEALER PLAYS
82 E(P)=E(P)-1
83 T(P)=T(P)-1Ø
84 IF V(1)=2 THEN 79
85 V(3)=V(3)+1
86 INPUT "HIT";V(1)
87 IFV(1)<>3 THEN 116
88 IF V(2)>Ø THEN 114
89 IF V(3)<>2 THEN 114
90 IF Q(S)=Q(G) THEN 93
91 PRINT"NOW IS THAT A PAIR?"
92 GOTO 86
93 V(2)=1
94 IF Q(G)<>1 THEN 96
95 V(1)=2
96 P=3
97 PRINT"PLAY HAND 1 NOW"
98 PRINT"FIRST CARD IS",
99 W(3)=W
100 X=G
101 GOSUB 103
102 GOTO 45
103 GOSUB 223
104 GOSUB 229:REM PRINT CARD
105 V(3)=1:T(P)=C
106 RETURN
107 P=2:V(2)=2
108 PRINT"PLAY HAND 2 NOW"
109 PRINT"FIRST CARD IS",
110 X=S
111 GOSUB 103
112 IF Q(G)=1 THEN 45
113 V(1)=Ø:GOTO 45
114 PRINT"NO SPLITS NOW"
115 GOTO 86
116 IF V(1)<>2 THEN 121
117 IF V(3)=2 THEN 12Ø
118 PRINT"TOO LATE TO DOUBLE"
119 GOTO 86
120 W(P)=W(P)*2
121 IF V(1)>Ø THEN 45
122 GOTO 79
200 '+++++SUBROUTINES+++++
201 '++++++++DEAL++++++++
202 GOSUB 207
203 T(P)=T(P)+C
204 IF V(5)=Ø THEN 206
205 V(5)=Ø:RETURN
206 GOSUB 229:RETURN
207 N=1Ø*(1+ABS(COS(N+W1)))
208 FOR A=1 TO N
209 X=INT(52.9999999#*RND(1))
210 IF X=Ø THEN 209
211 NEXT A
212 IF D(X)=Ø THEN 222
213 X=X+1:IF X>52 THEN X=1
214 R=R+1
215 IF R<5Ø THEN212
216 FOR A=1 TO 52
217 IF D(A) =K THEN 219
218 D(A)=Ø
219 NEXT A:R=Ø:PRINT
220 PRINT"**I RESHUFFLED**"
221 GOTO 207
222 R=Ø:D(X)=1
223 IF Q(X)<>1 THEN 226
224 C=11
225 E(P)=E(P)+1:RETURN
226 IF Q(X)>1Ø THEN 228
227 C=Q(X):RETURN
228 C=1Ø:RETURN
229 '++++++PRINT CARD++++++
230 GOSUB 232:GOSUB 241
231 RETURN
232 IF Q(X)<>1 THEN 234
233 PRINT"  ACE  ";:RETURN
234 IF Q(X)>1Ø THEN 236
235 PRINT Q(X);:RETURN
236 IFQ(X)>11 THEN 238
237 PRINT"  JACK  ";:RETURN
238 IFQ(X) > 12 THEN 24Ø
239 PRINT"  QUEEN  ";:RETURN
240 PRINT"  KING  ";:RETURN
241 '++++++PRINT SUIT++++++
242 IF X>39 THEN 246
243 IF X>26 THEN 247
244 IF X>13 THEN 248
245 PRINT"OF SPADES":RETURN
246 PRINT"OF CLUBS":RETURN
247 PRINT"OF HEARTS":RETURN
248 PRINT"OF DIAMONDS":RETURN
249 '++++++DEALER PLAYS++++++
250 P=2
251 PRINT"MY HOLE CARD IS ";
252 X=M
253 GOSUB 229
254 IFT(2)<22 THEN 257
255 IF V(2)=Ø THEN 268
256 IF T(3)>21 THEN 268
257 P=1
258 IF T(1)<17 THEN 272
259 IF T(1)>17 THEN 261
260 IF E(1)>Ø THEN 272
261 IFT(1)>21 THEN 275
262 P=2
263 IF T(P)>21 THEN 268
264 IF T(1)>21 THEN 267
265 IF T(1)>T(P) THEN 268
266 IF T(1)=T(P) THEN 269
267 W1=W1+W(P):GOTO 269
268 W1=W1-W(P)
269 IF V(2)>Ø THEN 271
270 GOSUB 282:GOTO 23
271 P=3:V(2)=Ø:GOTO 263
272 PRINT"I DRAW  ",
273 GOSUB 201:GOTO 257
274 GOTO 257
275 IF E(1)=Ø THEN 279
276 E(1)=E(1)-1
277 T(1)=T(1)-1Ø
278 GOTO 257
```

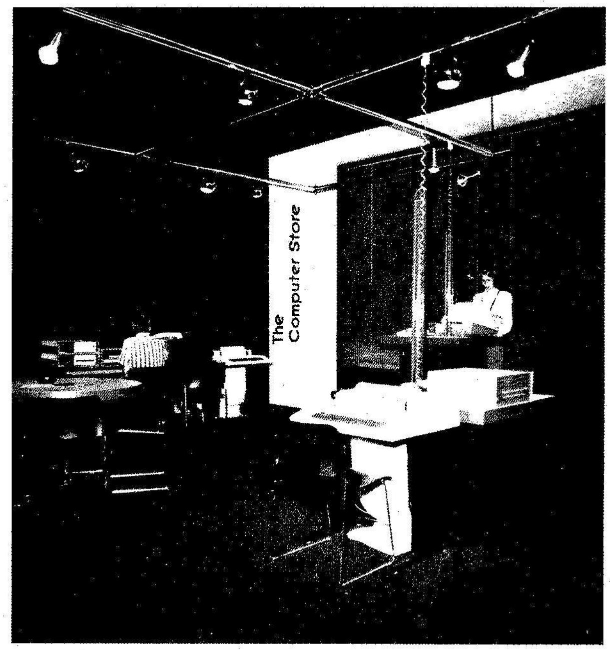
Continued on Page Nine

the sound and the fury

Photographs by Jerry Waters
of the **Charleston Gazette**
Tuesday, February 1, 1977

CHARLESTON, W. VA.—A new entrance to the downtown Charleston Municipal Parking Building was created February 1 when this car smashed through a wall of The Computer Store on the Quarrier Street side of the building. The driver of the car, Carl Davis, 48, of Crescent Road, escaped injury in the smashup but was charged with driving while intoxicated. City Building Commissioner Roy Jones surveyed the damage and found that the parking building was still safe to use, since the accident resulted in no major structural damage.





HAM on the side

Altair SIOB Makes Ham Interfacing Easy

By David Le Jeune

More and more ham operators have caught the computer bug and want to interface their Altair 8800 computers to a ham RTTY or CW station. However, since the Federal Communications Commission (FCC) regulates the design of all ham stations, not just any interface board can be used.

The FCC restricts amateur RTTY stations to 5-level asynchronous Baudot coded transmissions. The use of cyphers or codes are prohibited (i.e., all transmissions must be in plain text), and the speed of transmissions must not exceed 75 baud. The most commonly used baud rate is 45.45 baud (60 words per minute for 7.2 unit code - 1 start bit, 5 data bits and 1.2 stop bits). The FCC also requires ham station operators to end each transmission by sending the transmitting station's call sign in CW.

The Altair SIOB interface board is the ideal choice to meet these requirements. With minor modifications, the Altair SIOB can be strapped to provide both the 45 baud, 5 level, 7.5 or 8 unit RTTY port. The CW port and a transmitter on-off control port.

Modifications to SIOB

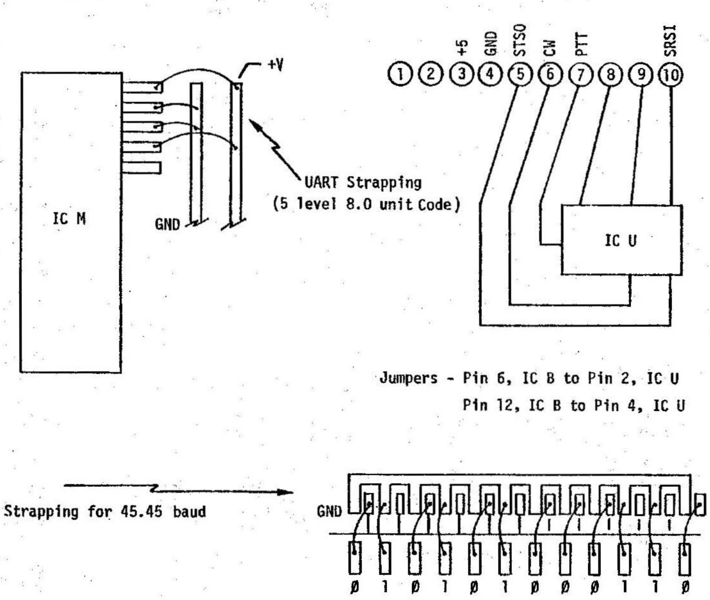


Figure 1

The SIOB board uses the COM 1602 UART. This IC (IC M) must be strapped to provide 2 stop bits (NSB high-pin 36), 5 data bits (NDB1 and NDB2 low-pins 37 and 38) and no parity (NPB high-pin 35). In order to derive a 45.45 baud clock (45.45 baud clock (45.45 * 16 = 727.2 Hz), the divide by N counter (ICs P, Q and R) must be strapped as shown in Figure 1. This strapping provides a 727.2 Hz clock to the UART, IC M. The only other modification to the board is to run jumpers from IC B (74L00), pin 6 to IC U (8T97), pin 2 and from IC B, pin 12 to IC U, pin 4. The board can be addressed for any port assignment. But the software, which will be described in future C.N. articles, will be written for the Baudot I/O at port 24_8 (Status/Control) and 25_8 (Data). (Note: The Altair 88-2SIO cannot be used for two reasons. The Motorola ACIA chip used on the board cannot generate a 5-level, 7.2, 7.5 or 8.0 unit code. Secondly, the on-board clock cannot be set to 45.45 baud, and the ACIA cannot be programmed to provide the baud rate with the choice of clock frequencies available.)

Continued

"Hit Me Again"
Blackjack Program continued

```
279 PRINT"***I BUSTED***";
280 GOTO 262
281 '++++++PAY UP++++++
282 IF W1<0 THEN 286
283 IF W1=0 THEN 288
284 PRINT"YOU'RE AHEAD $"W1
285 RETURN
286 PRINT"YOU'RE BEHIND $";-W1
287 RETURN
288 PRINT"YOU'RE EVEN":RETURN
289 END
OK
```

ASDC, AUG Functions Outlined

Continued

QUESTION	ASDC	AUG
7. To which organization should I submit my software?	Submit your software to the ASDC if you are experienced at programming and documentation and feel that the software system you have created has commercial value and can successfully undergo intensive testing and evaluation. Small programs and subroutines should not be submitted to the ASDC.	Submit your software to the AUG if you are an interested, competent programmer who would like to see your program made available at a nominal charge to all other Altair computer users.

The following is a description of the modification I made to the RTTY terminal unit (modem) I use in my ham station. The unit, one of the most popular RTTY terminal units available, is the ST-6, designed by Irv Hoff W6FFC, and first described in HAM RADIO magazine, January 1971. Figure 2 is a partial schematic of the ST-6. It shows the "slicer" stage of the unit and the selector magnet driver--an MJE 340 high voltage transistor that keys the 60 ma 110VDC TTY selector magnet current loop, typical of most TTY machines in use on amateur RTTY. The line between the cathode end of CR14 and the 2.2K resistor (R25) is broken, and jumpers are brought out to any convenient tie point. The received signal at point A is a close approximation to an inverted RS-232 signal with mark at +12 volts and space at -12 volts. This has to be converted to TTL levels before sending it to the UART on the SIOB interface board. Signals from the Altair computer will be tied into the ST-6 selector magnet circuit at point B.

Additional interface circuitry is shown in Figure 3. Q2 converts the incoming signal to TTL levels. It is then inverted by U1B and then sent to the SIOB board. IC U1A allows either the serial data stream from the UART or the CW output port to key the selector magnet loop. Most amateur RTTY stations use this method for normal RTTY and CW Identification purposes. However, for standard on-off keying purposes, the signal from the CW output port is also fed to IC U2B--a 7406 that is used to drive a reed relay. The transmitted on-off control port signal is fed to another 7406 (IC U2C), which drives a reed relay in the transmitter push-to-talk (PTT) line. The reed relays I used are small ones that draw about 20 ma.

Next month I'll cover the software requirements for this interface.

ST-6 Modifications

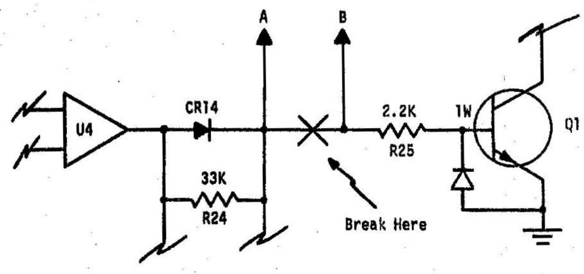


Figure 2

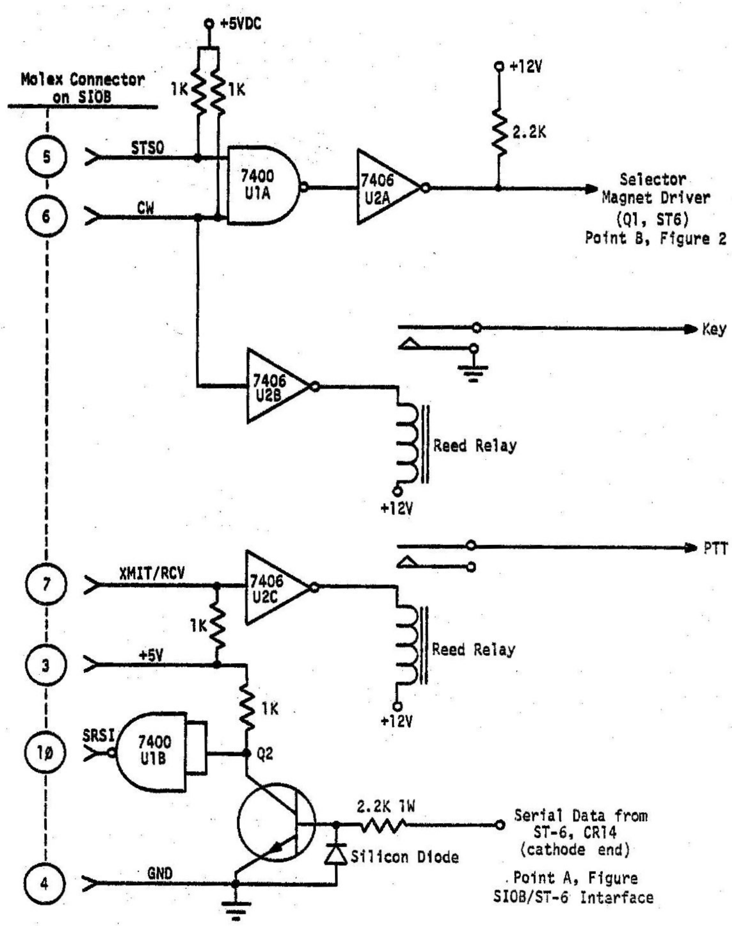


Figure 3

GLITCHES

Q & A from the Repair Department

Simple Fix Developed For Altair 8800A Power Supply Problems

BY: Bruce Fowler

The Altair 8800A power supply is rated to provide +8 volts pre-regulated at 8 amps. Since all of the MITS circuit boards were initially designed to draw .5 amp each, this met all the requirements for a 16-slot motherboard. The motherboard has since been expanded to 18 slots, and some boards, like the Altair 4K Static Memory board, draw considerably more than .5 amp. In a fully loaded chassis, this means that the +5 volt supply could be overtaxed.

This situation can be detected by examining pin 1 or pin 51 on the bus with a scope. The 7805 regulators used in the Altair 8800A require a minimum of 7 volts at the ripple trough. The regulators also have a thermal shutdown circuit that activates if too much current is drawn. For these reasons, the average supply to each regulator should be about 9 volts.

If the scope shows that the 7805 input is below 7 volts at the

Continued

ripple trough, the regulator may not work properly. If this is the case, the following modifications should be made to the power supply. The collector and emitter leads of the TIP 140 (or 141) Darlington pair should be shorted. This is most easily done by connecting lugs 5 and 8 on the terminal block (where lug 1 is at the top). (See Figure 1.) There is a two volt drop across this Darlington pair, and this jumper effectively bypasses the TIP 140, providing more voltage to the bus.

As long as six or more boards are installed in the chassis, the voltage should not go above 9 volts. If fewer cards are installed, however, the jumper should be taken out.

On one Altair test chassis, we solved the problem of varying numbers of boards by putting a switch across the Darlington pair from collector to emitter. This permitted us to switch the Darlington pair in and out of the circuit as required.

More severe power supply limitations may be encountered by owners of the first Altair 8800 models who have more than six boards. A higher voltage power transformer will be provided to any Altair 8800 owner who also owns six MITS circuit boards. This should eliminate any power supply deficiencies. For more information about obtaining this power supply modification kit, contact your local Altair Computer Center.

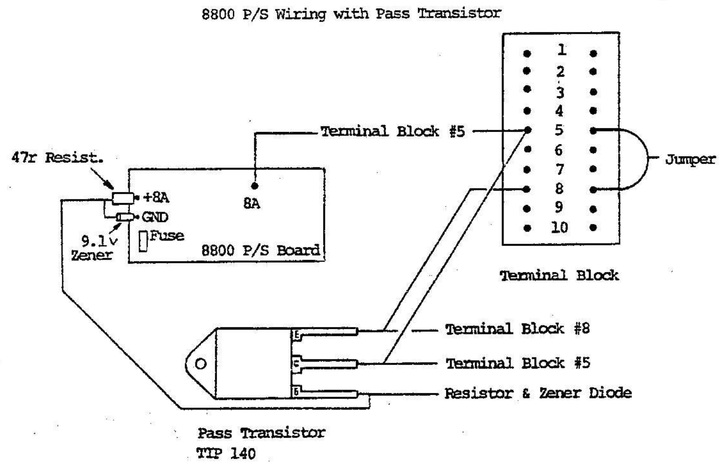


Figure 1

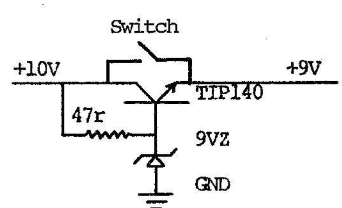


Figure 2

BITS AND PIECES

By Sondra Koppenheffer

MITS Repair Policy:

The following procedure should be followed when returning an item to MITS for repair:

1. Protect the item with sufficient packing materials to prevent damage during shipment.

2. Include a letter with the following information inside the box:

   a. Your name, address and phone number.

   b. A list of the items returned AND their MITS part number.

   c. A description of the problem(s).

   d. Warranty expiration date. If the warranty has already expired on an item, indicate your payment method for repair charges.

   e. Where and when you purchased the item(s). For items purchased directly from MITS, please indicate the original order number of the item.

   f. Any other name under which your customer files might be located.

3. Insure the package for the original amount for which it was purchased.

4. After returning an item, allow 2-3 weeks before contacting MITS again.

By carefully following this procedure, repairs can be done more quickly and effectively. However, the more research an item needs, the more time it takes for repairs.

Information By Telephone:

Telephoning the Marketing Department is an easy way to have your questions answered or to check the status of your order. We're always more than happy to help. But we can locate the required information more quickly and keep such long distance calls to a minimum if everyone would keep the following guidelines in mind.

Continued

BITS AND PIECES
Continued

1. Tell the receptionist the specific area of the Marketing Department with which you want to be connected (a list of the department heads for each area is provided below). This will save time in transferring calls. Also, please indicate to the receptionist if you are with a company or university.

2. If you are checking on the status of an order, please have the following information ready when you are connected with the correct department.

   a. Exact items ordered
   b. The MITS order number
   c. The purchase order number (industrial and international sales only)
   d. The name under which your file can be located
   e. The date your order was sent to MITS

3. Please follow these same guidelines if you send an order by mail.

Vice President of Marketing:	Pam Holloman
Secretary to Vice President:	Elva Tapia
Industrial Sales:	Dorothy West
International Sales:	David Ning Lucia Wilcox
Hobbyist Sales:	Sondra Koppenheffer Patti Montoya
Applications Engineer:	Rich Haber
Public Relations:	Chuck Olsen
Marketing Staff Assistant:	Kris Ray

Altair Users

Don Chamberlain
9457 Las Vegas Blvd. South #321
Las Vegas, Nevada 89119
Phone: (702) 361-4924

Alfred R. Howes
Box 342 Boyce Rd.
Glenford, NY 12433

Dick Fehriback
5779 Blaine SE
Grand Rapids, MI 49508
(616) 455-3138

UNIVERSAL I/O INCREASES 680'S VERSATILITY

By Randy Huddleston

The Altair 680b Universal I/O board is a valuable addition to the 680b system, because it allows the 680b user to take advantage of video terminals, line printers and parallel I/O devices as well as serial teleprinters. Since the software of the 680b system is designed to communicate through the main serial I/O port, however, some software changes must be made before the full potential of the UIO can be realized.

The Altair 680b Monitor PROM is the key to the computer's ease of use. It contains all the necessary routines for loading programs, examining and modifying memory and starting programs at any address. It is also, unfortunately, the source of the difficulty in using the new UIO serial and parallel ports, because the Monitor only contains the address of the 680b's main serial port and thus, can only load programs through that port.

There are two ways around this problem. One method, the most useful in theory but the most difficult in practice, is to reprogram the ACIA PROM to change the I/O addresses. This would allow the computer to communicate through any port. Such reprogramming is, of course, impossible without a PROM programmer. But if a programmer is available, the ACIA PROM can be changed according to the information in Table 1, which applies to the UI/O serial port.

The second way to circumvent the port addressing problem only works in BASIC, but it can be done without reprogramming any PROMs. In essence it involves duplicating the CONSOLE function used in Altair 8800 Extended BASIC. With the CONSOLE function, control of BASIC can be shifted from one I/O port to any other port. To implement this function, a program must be written to simulate the I/O scheme of the 680b's ACIA with the I/O addresses changed to access the new port. This program can be stored in PROM or at some location in high (above BASIC) memory. Then BASIC must be modified to look for the I/O routine there instead of in the Monitor. BASIC must also be changed so that the new I/O port's control register is loaded with the correct initialization information.

The remainder of this article will discuss the procedure for using the UIO's serial port. The procedure for setting up the parallel ports is virtually identical, except for different initialization bytes and port addresses.

Table 2 shows the new I/O subroutine. If the routine is to be programmed onto a PROM, it should start at location FCØØ. If it is to be loaded into RAM, it should start at location 3FØØ. But this can change depending upon the amount of memory used.

Five locations in BASIC must be modified in order to transfer control to the new subroutine. These locations can be modified before or after BASIC has been initialized. Table 3 shows the changes needed to transfer control to the new subroutine. Table 4 shows the changes needed to transfer BASIC control back to the monitor subroutine.

The port is most easily initialized by using the monitor's "M" command to load the ACIA initialization byte. This routine first does a master reset and then loads the control register with the proper status information.

The final listing (see page) is a complete version of the CONSOLE function. It is designed to reside on a PROM that starts at location FCØØ. The routine does all port initializations and an automatic CONSOLE by jumping to different locations. It modifies BASIC as it runs. After BASIC has loaded, the normal procedure is to jump to address zero. In this case if no transfer is wanted, a jump to zero will respond as if nothing changed. A jump to FC21 will bring up BASIC on the UI/O port, and a reset followed by a jump to FC47 will return control to the main port. Again, except for the initialization procedure, this system works the same for the parallel ports.

By using the modified monitor PROM to work through the UI/O board, BASIC can be loaded, followed by the PUNBAS program (see p. 27, Nov., COMPUTER NOTES). Then BASIC can be saved on cassette by switching the port from TTY to RS-232 and writing through an audio cassette interface. Loading BASIC from cassette should be considerably faster than loading from paper tape.

Table 1

Address	Change To
FF0A	07
FF26	06
FF8D	07
FFE1	06
FFED	06

Continued

Table 2

PROM Address = FC00		16K Address = 3F00	
INCH	BSR POLCAT	8D	0C
	BCC INCH	24	FC**
	LDA B#$7F	C6	7F
	CMP B#$F3	D1	F3
	ANO B F007	F4	F0 07*
	BCC OUTCH	24	06
	RTS	39	
POLCAT	LDA B F006	F6	F0 06*
	ASR B	57	
	RTS	39	
OUTCH	FCB $8C ; SKIP TRICK	8C	
OUTS	LDA B #$20	C6	20
	PSH B	37	
OUTCI	BSR POLCAT	8D	F5
	ASR B	57	
	BCC OUTCI	24	FB
	PUL B	33	
	STA B F007	F7	F0 07*
	RTS	39	

Table 3

BASIC MODIFICATIONS

FROM MONITOR TO UIO

If 16K:		If PROM:	
0420	3F	0420	FC
08AE	3F	08AE	FC
08AF	13	08AF	13
0619	3F	0619	FC
061A	0E	061A	0E

Table 4

BASIC MODIFICATIONS

FROM UIO TO MONITOR

0420	FF
08AE	FF
08AF	81
0619	FF
0620	24

Continued

VTL-2 Now Offered for Altair 680b Computer

VTL-2 is a very tiny language developed for the Altair 680b computer. It is both a simple language interpreter(similar to BASIC) and a collection of useful subroutines for the machine language programmer. VTL-2 resides completely in Read-Only-Memory. It uses various subroutines in the MITS 680 ACIA monitor.

VTL-2 is designed for use with a minimal system of 1024 bytes of Random Access Memory. However, the language can use all available memory.

In addition to being a very useful language in its own right, VTL-2 is supplied with a complete source listing, so that the user has a complete set of fully-documented utility subroutines which can be used by machine-language programs even when the VTL-2 interpreter is not being used.

VTL-2 employs standard BASIC line correction and backspacing facilities. Lines may be added, deleted or changed by number, providing program editing flexibility.

All arithmetic in VTL-2 is in 16-bit integer mode. One special variable called "%" contains the remainder after division operations, easing the implementation of multiple-precision subroutines.

VTL-2 has one array, which is as large as memory will allow. It can be broken down into several sub-arrays for flexibility. VTL-2 will print strings as well as input and output them as single-character variables. Longer strings can also be stored in the array.

The 768 bytes of PROM memory required for VTL-2 is less than half of that required by the next smaller high-level language interpreter. Keeping this in mind may help the user to understand some of the language's limits as compared to full BASIC. We trust that you will be pleasantly surprised to discover just how much computing power can be squeezed into a tiny space!

The VTL-2 package comes complete with programmer's manual, a copy of the source listing and some sample games that can be played with only 1K of RAM memory.

VTL-2 is avaiable for $114(postpaid) from The Computer Store, 820 Broadway, Santa Monica, California 90401, or from any other Altair Computer Center.

ALTAIR ACR OPERATION EASY TO LEARN

By Rich Haber

First in a two-part series on the theory of operation and repair of Altair ACR cards.

The Altair Audio Cassette Recorder (ACR) card's ability to provide large data storage and program loading through an audio cassette recorder has made it a very popular interface. Many readers have requested more information on the ACR so that they can service their own boards. This series, which will cover theory of operation in part one and repair in part two, should answer many of your questions.

Theory of Operation

The 88-ACR consists of two boards which allow an inexpensive cassette recorder to be used as a mass memory storage device. The 88-ACR can read or store data on an audio cassette by recording different frequency tones for the two logic levels. A 2400 Hz tone represents a logic "1," and an 1850 Hz tone represents a logic "0." When the tape is played, it outputs a signal of changing frequency, which is similar to frequency shift keying (FSK) transmission. However, the computer cannot use this data because it can only interpret TTL level parallel data. The ACR uses two separate boards to convert the information on the tape into the correct format. The Modem board converts the serial fm data to serial TTL level digital data. The SIOB board then converts it to parallel data. Each board will convert data in either direction. (see Figure 1, opposite page)

The Modem Board consists of two parts: the modulator section and the demodulator section. The modulator section takes serial digital data from the SIOB and modulates it into an fm audio signal that can be recorded onto a tape. The demodulator section inputs data from the tape deck, demodulates the fm signal into digital data and presents it to the SIOB. (See Figure 2, page 16.)

The demodulator section is represented on the top third of the Modem schematic. (See Figure 3.) Data from the tape deck enters from the top left at FSK Play Input. It is filtered through a band-pass and fed into an op-amp (IC A). The output of A is filtered again through another band-pass and fed into another op-amp (IC B). The output of B is then fed into IC C--the actual demodulator chip.

Continued

UNIVERSAL I/O INCREASES 680'S VERSATILITY

Continued

Table 5

ACIA INITIALIZATION

.M	F006	XX	03
.M	F006	XX	81

680 CON/C — CONSOLE PROM — PROM LOC: FC00

ADD HEX	ADD OCTAL	HEX	OCTAL	MNEMONICS		
00	000	8D	215	INCH	BSR	POLCAT
01	001	0C	014			
02	002	24	044		BCC	INCH
03	003	FC	374			
04	004	C6	306		LOAB	#$7F
05	005	7F	177			
06	006	D1	321		CMPB	#$F3
07	007	F3	363			
08	010	F4	364		ANDB	F007
09	011	F0	360			
0A	012	07	007			
0B	013	24	044		BCC	OUTCH
0C	014	06	006			
0D	015	39	071		RTS	
0E	016	F6	366	POLCAT	LOAB	F006
0F	017	F0	360			
10	020	06	006			
11	021	57	127		ASRB	
12	022	39	071		RTS	
13	023	8C	214	OUTCH	FCB	$8C
14	024	C6	306	OUTS	LDAB	#$20
15	025	20	040			
16	026	37	067		PSH B	
17	027	8D	215	OUTC1	BSR	POLCAT
18	030	F5	365			
19	031	57	127		ASR B	
1A	032	24	044		BCC	OUTC1
1B	033	FB	373			
1C	034	33	063		PUL B	
1D	035	F7	367		STA B	F007
1E	036	F0	360			
1F	037	07	007			
20	040	39	071		RTS	
21	041	86	206		LDAA	MR
22	042	03	003			
23	043	B7	267		STAA	F006
24	044	F0	360			
25	045	06	006			
26	046	86	206		LDAA	
27	047	B1	261		÷16 8 bits	
28	050	B7	267			
29	051	F0	360		STAA	F006
2A	052	06	006			
2B	053	86	206		LDAA	FC
2C	054	FC	374			
2D	055	B7	267		STAA	0420
2E	056	04	004			
2F	057	20	040			
30	060	86	206		LDAA	FC
31	061	FC	374			
32	062	B7	267		STAA	08AE
33	063	08	010			
34	064	AE	256			
35	065	86	206		LDAA	13
36	066	13	023			
37	067	B7	267		STAA	08AF
38	070	08	010			
39	071	AF	257			
3A	072	86	206		LDAA	FC
3B	073	FC	374			
3C	074	B7	267		STAA	0619
3D	075	06	006			

Continued

Table 5
Continued

680 CON/C — CONSOLE PROM — PROM LOC: FC00

ADD HEX	ADD OCTAL	HEX	OCTAL	MNEMONICS	
3E	076	19	031		
3F	077	86	206	LDAA	0E
40	100	0E	016		
41	101	B7	267	STAA	061A
42	102	06	006		
43	103	1A	032		
44	104	7E	176	FMP	0000
45	105	00	000		
46	106	00	000		
47	107	86	206	LDAA	MR
48	110	03	003		
49	111	B7	267	STAA	F000
4A	112	F0	360		
4B	113	00	000		
4C	114	86	206	LDAA	B1
4D	115	B1	261	÷16	
4E	116	B7	261	STAA	F000
4F	117	F0	360		
50	120	00	006		
51	121	86	206	LDAA	FF
52	122	FF	377		
53	123	B7	267	STAA	0420
54	124	04	004		
55	125	20	040		
56	126	86	206	LDAA	FF
57	127	FF	377		
58	130	B7	267	STAA	08AE
59	131	08	010		
5A	132	AE	256		
5B	133	86	206	LDAA	81
5C	134	81	201		
5D	135	B7	267	STAA	08AF
5E	136	08	010		
5F	137	AF	257		
60	140	86	206	LDAA	FF
61	141	FF	377		
62	142	B7	267	STAA	0619
63	143	06	006		
64	144	19	031		
65	145	86	206	LDAA	24
66	146	24	044		
67	147	B7	267	STAA	061A
68	150	06	006		
69	151	1A	032		
6A	152	7E	176	JMP	0000
6B	153	00	000		
6C	154	00	000		

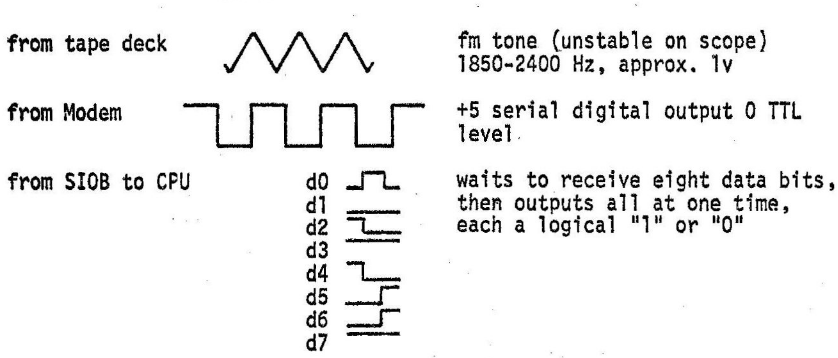


Figure 1

ALTAIR ACR OPERATION EASY TO LEARN

Continued

For FSK demodulation, the circuit is connected as a PLL system by ac coupling the VCO output (pin 15) to pin 6. The FSK input is applied to pin 4. When the input frequency is shifted, corresponding to a data bit, the polarity of the dc voltage across the phase detector outputs (pins 2 and 3) are reversed. The voltage comparator and the logic driver section convert this dc level shift to a binary pulse. One of the phase detector outputs (pin 3) is ac grounded and serves as the bias reference for the voltage comparator section. Capacitor C17 serves as the PLL loop filter, and C16 and C15 serve as post-detection filters. The timing capacitor, C14, and the fine-tune adjustments are used to set the VCO frequency, f_o, midway between the "mark" and "space" frequencies of the input signal.

The output of the demodulator chip at pin 8 uses a current sink logic. This means that when the output is a logic "0," the chip will ground the line and drop the voltage to zero. When the output is a logic "1," pin 8 will be high impedance, thereby allowing the power supply to pull the line up to +5v. Thus, R32, shown on the extreme right of the Modem diagram, is called a "pull-up" resistor. It is connected to Vcc, and data is jumpered from RS Play Serial Data on the Modem to the $\overline{RSI}$ input on the SIOB board.

SIO stands for Serial Input/Output. (See Figure 4.) The B is our code, and it means the board is TTL (transistor-transistor logic) compatible. This IC is a UART (Universal Asynchronous Receiver-Transmitter). It has the ability to receive serial data, reformat it and output it as parallel data on individual data lines or vice-versa. This IC is very flexible and has many inputs to tell it how it should format the data. It has to be told the number of data bits to receive, the number of stop bits, parity or not parity, etc. (Parity is a system for checking the accuracy of transmitted data, but we do not use it at MITS.) One stop bit, which indicates the end of a transmitted byte, is used for the ACR. There are eight data bits.

The UART has four control inputs:

16-$\overline{SWE}$ - Status Word Enable--allows the status of the UART to be output on the data lines to the CPU.

4 $\overline{RDE}$ - Received Data Enable--tells the UART to output the received data to the CPU.

Continued

ALTAIR ACR OPERATION EASY TO LEARN

Continued

23 $\overline{TDS}$ - Transmit Data Strobe--this signal tells the UART to take data off the bus and transmit it serially.

18 $\overline{RDAV}$ - Reset Data Available--resets the data available flip-flop while the CPU is receiving the data.

Logic gates J, G and S control the above commands. The CPU tells the UART if it wants to input or output data by the control signals at left center.

SINP - the CPU wants to input data

SOUT - the CPU wants to output data

$\overline{PWR}$ - data on the bus is valid and should now be transmitted.

Every I/O board has two channels. The odd channel is used for data and the even channel is used to tell the CPU the existing conditions in the UART. Both channels use the same data lines. The ACR uses channels 6 and 7.

IC I is an eight-input NAND gate that enables the logic section when address 6 or 7 is on the bus. When strapping the address section, each address line is sent through an inverter when it is set low. When set high, it bypasses the inverter. Therefore, if A7 through A3 are inverted and A2 and A1 are direct when address 006 or 007 is on the bus, all inputs to NAND gate I will be high. Pin 8 will go low and partially enable J1 and J4. A0 is used to select the odd channel (data) or even channel (control), depending on whether it is high or low. The four commands to the UART can be produced by the logical gating of A0, SINP, SOUT and $\overline{PWR}$, as shown on the SIOB schematic. These inputs also control the enable/disable functions for the buffers on the DI (data into the CPU) lines.

The bottom third of the schematic deals with the interrupt capability of the board. This circuitry permits interruption of the CPU when data is input and/or output. These interrupts can be given a relative priority by the strapping shown at the bottom right of the SIOB schematic. Interrupt capabilities (lower left portion of the schematic) are under software control.

As mentioned before, the SIOB has a control channel that is used to tell the CPU what its current function is and to identify any transmission errors. Each of these indicators is sent over a specific data line when the control channel is called and when $\overline{SWE}$ (Status Word Enable) is low. The bit definition of these outputs is shown below.

Continued

Data Bit	Logic Low Level	Logic High Level
0	Input device ready (Data is available for computer to input)	Not ready
1	Not used	Not used
2		Parity error
3		Framing error (data word has no valid stop bit)
4		Data Overflow (a new word of data has been received before the previous word was input to the accumulator)
5	Not used	
6	Not used	
7	Output device ready (Transmitter buffer is empty.) Interrupt to occur if interrupt is enabled.	Not ready

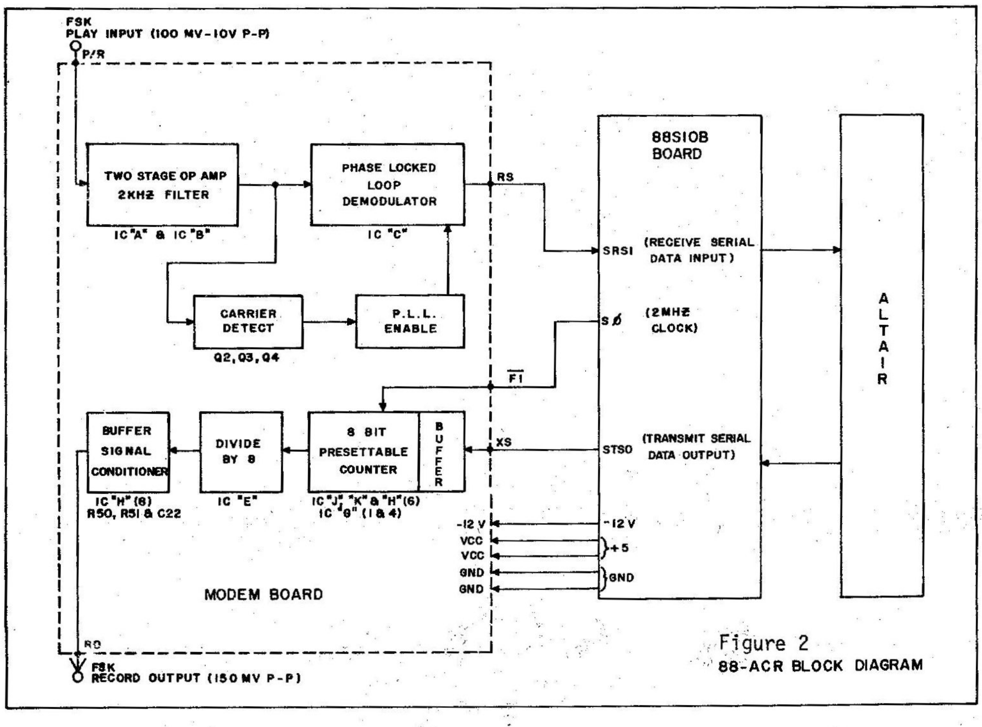


Figure 2
88-ACR BLOCK DIAGRAM

Continued

ALTAIR ACR OPERATION EASY TO LEARN

Continued

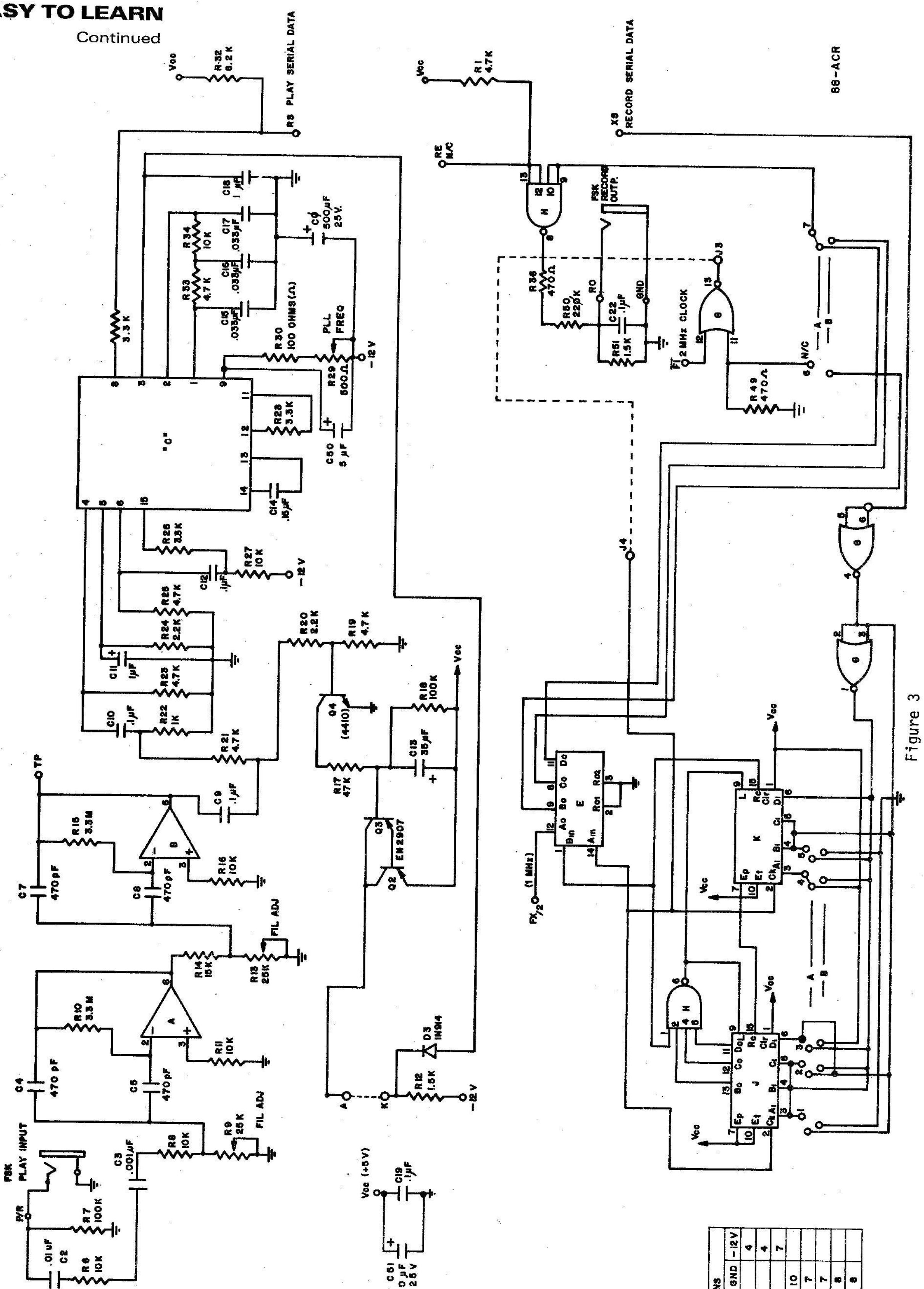


Figure 3

POWER CONNECTIONS				
IC	TYPE	V_{CC1}	GND	−12V
A	741	7		4
B	741	7		4
C	XR 210	16		7
E	7493	5	10	
G	7402	14	7	
H	7420	14	7	
J	9316	16	8	
K	9316	16	8	

ACR-3

ACR-1

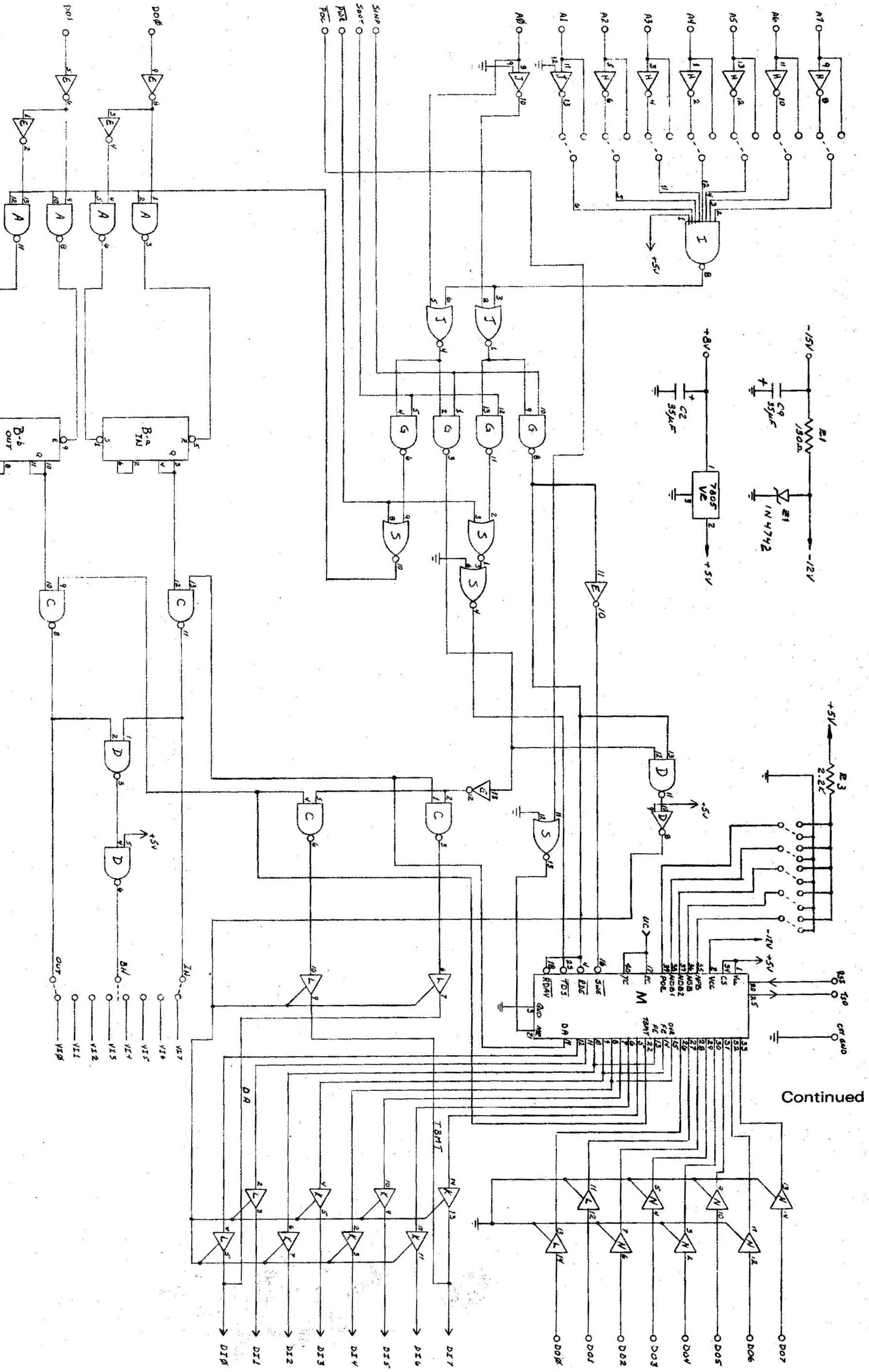


Figure 4

Continued

ALTAIR ACR OPERATION EASY TO LEARN

Continued

Inputting Data to the SIOB

Serial data from the Modem appears at the RS1 input (pin 20 of the UART). The input at pin 20 is normally high. When it goes low, a counter (which counts from 1 to 16) is started. This is provided by a clock input at pins 17 and 40. This clock is a negative pulse occurring at 16 times the baud rate. If the input at 20 is still low after eight clock periods (halfway through the incoming start bit), the clock will interpret it as a valid start bit. From then on, each bit is sampled on the eighth clock pulse and loaded into a shift register.

After eight bits have been received, it looks for a parity bit, then for a stop bit. If the stop bit is not present, it sets a framing error flag. When the register is full, the data is sent to an output holding register. The Data Available flag goes high, telling the CPU it is ready to send data. The CPU then issues a SINP signal and calls the data channel. This sends $\overline{\text{RDE}}$ and $\overline{\text{RDAV}}$ (pins 4 and 18) low, and the UART outputs the data to the CPU.

Outputting Data Through the SIOB

When the computer is turned on, the $\overline{\text{POC}}$ clears the registers through pin 21 (master reset) and puts the UART into an idle state. When the UART is ready to input a byte for transmission, it will set pin 22 TBMT (Transmit Buffer Empty) high. When the CPU sees this, it will output data onto the DO lines. The CPU sends out a SOUT and $\overline{\text{PWR}}$ signal and calls the data channel. These signals are gated to become a negative going pulse to pin 23 ($\overline{\text{TDS}}$, Transmit Data Strobe). On the leading edge of this pulse, the UART will input data from the DO lines to a holding register. On the trailing edge of the pulse, the data is moved to the transmitter register where the start and stop bits are added and transmission is started. The data is output as serial TTL level data from pin 25 (TSO, Transmitter Serial Output). This output appears at pin 5 of the molex connector. From there it is jumpered to the XS Record Serial Data input to the Modem.

Recording Data Through the Modem

The modulator section of the Modem is shown on the bottom third of the Modem schematic. (See Figure 3.) The input labelled $\overline{\text{FI}}$ is a 2MHz clock input. The Modem divides this frequency down to 2400 Hz to indicate a logic "1" or to 1850 Hz to

Continued

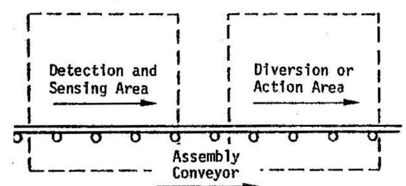
COMPUTER INSPECTS INDUSTRIAL OUTPUT

By Bill Kuhn

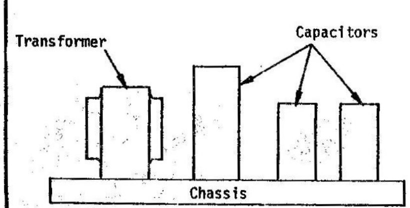
Most coverage of computer applications tends to highlight the exotic and the complex. But the fantastic potential for computers to take over many of the simple, mundane tasks of life are often overlooked. With the introduction of the Altair Process Control Interface boards, (88-PCI and 680b-PCI, see October and Jan.-Feb. issues of C.N.), very flexible, low-cost controllers for various industrial tasks can now be configured, using the Altair 8800b and 680b computers.

In an industrial production line environment the Altair computer can sort, inspect, and test assemblies; reject and/or mark the assemblies according to test results; and log the results.

The following example illustrates some of the hardware and software requirements of an assembly line inspection and sorting system. This system will detect the presence or absence of parts on an assembly, check their relative position and divert the path of the assembly along with the line according to conditions it senses.

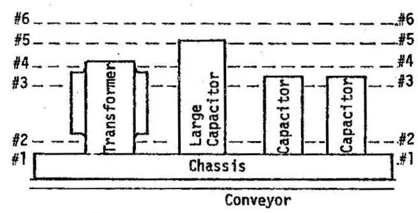


In this system let's assume that the assembly is a chassis with four additional parts mounted to its top. These parts might be a transformer and three electrolytic capacitors. (See Figure 2.) Let's also assume that all the chassis are aligned to the same attitude as they pass the detection and sensing area. A more complex system could be configured to recognize the assembly in any alignment, but aligning them the same way keeps things simple.



The assembly is scanned by a group of six photo detectors coupled to columned light sources across the path of the assembly conveyor. The photo detectors are each at a different height so that the components and the chassis itself break some of the light beams as the assembly passes on the conveyor.

The photo detectors and light beams are aligned at the following heights.



#1 is aligned above the conveyor and below the top of the chassis. This detector would sense the presence of the chassis itself and any other objects on the assembly conveyor.

#2 is just above the chassis top to detect the components mounted to the top of the chassis and any foreign objects on the chassis.

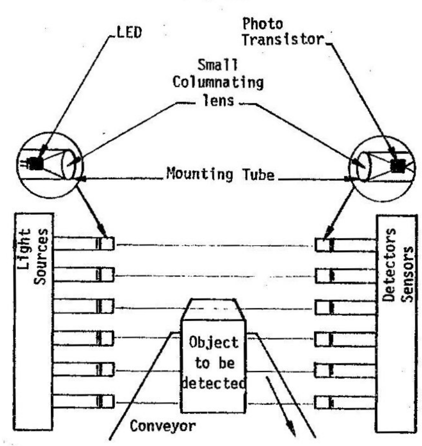
#3 is placed just below the top of the smaller capacitors.

#4 is just above the smaller capacitors but below the top of the transformer.

#5 is above the transformer but below the top of the larger capacitor.

#6 is just above the top of the larger capacitor.

Figure 4 shows how the sensor system looks from a viewpoint parallel to the conveyor.

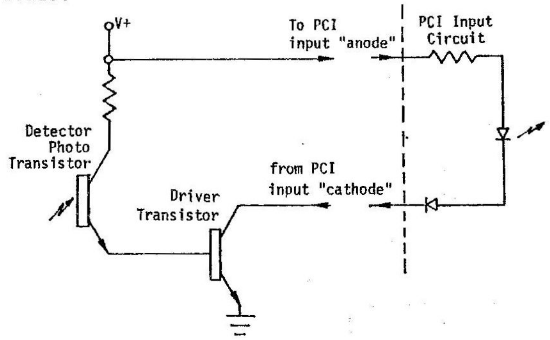


Continued

COMPUTER INSPECTS INDUSTRIAL OUTPUT

Continued

The photo detectors are connected to the inputs of the PCI board so that light falling on the photo transistor causes a current to flow through the input circuit of the PCI board.



So, the absence of any object breaking a particular light beam is "seen" by the computer as a logic LOW at the PCI input. The presence of some object causes a logic HIGH at the PCI.

The computer waits for detector #1 to show presence of the chassis. It then begins to clock in data from the sensor array (#1 through #6) and compare it with stored data in memory. In this way the computer "looks" at the profile of the assembly and compares it with a profile stored in memory. If the computer "sees" no difference in the incoming profile and its memory profile of a good assembly, the assembly is considered good or passing and is allowed to continue on the conveyor to its destination--the next assembly station, stockroom or shipping.

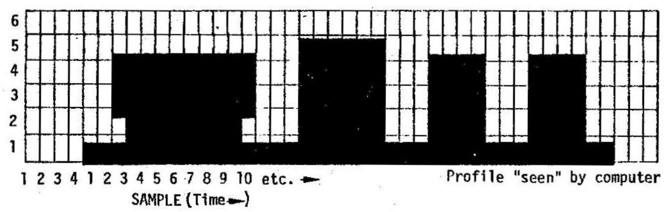
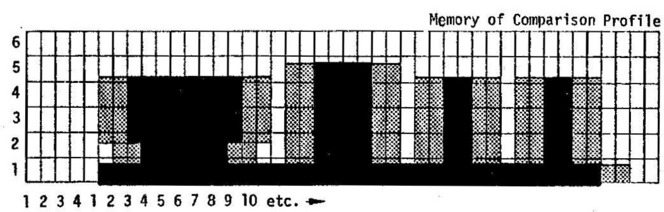


Figure 6 shows how the profile might appear to the computer. It represents a bit of data showing the presence of some object between the light source and the detectors as a blacked out square and the absence of an object as an open square. The vertical axis of the diagram in Figure 6 shows the data from the six photodetectors. The horizontal axis represents dimensional data as clocked into the computer with time.



The program that compares the input profile with the stored profile has to make allowance for areas of uncertainty at the edge of a sensed object. Depending on the accuracy (speed) of the samples, this uncertainty might be from one to three samples at each edge. Such tolerance can be arranged by masking out a couple of bits at the expected edges so that no comparison can be made in those areas of uncertainty. Figure 7, which shows the composite of a stored profile and a stored uncertainty mask, illustrates this principle. The blackened areas are compared for presence of an object, open areas for absence of an object, and gray areas are masked out so no comparison is made.

Continued

ALTAIR ACR OPERATION EASY TO LEARN

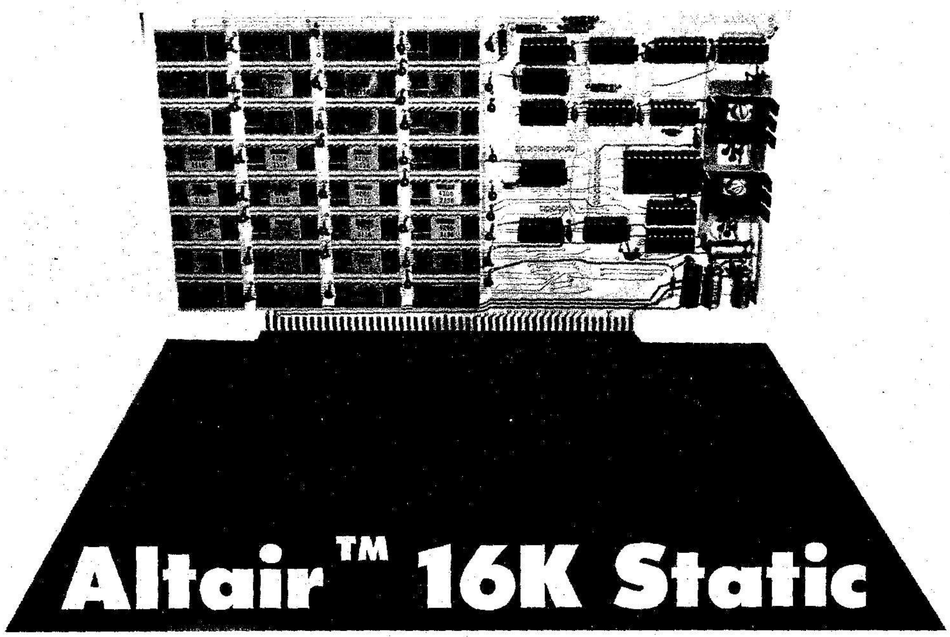
Continued

indicate a logic "0." The 2MHz signal is presented to the clock inputs of ICs J, K and E. Note that J3 and J4 are the jumper connections that have to be hooked together. Data from the UART appears at XS on the extreme right. The two NOR gates labelled G are used as inverters. The first gate supplies $\overline{\text{data}}$; the second supplies data. ICs J and K are synchronous 4-bit counters. Since the carry output of J is connected to the enable input of K, they can be thought of as eight flip-flops in series. Since each flip-flop will divide the signal by 2, eight in series will divide the signal by 256. IC E is strapped to divide by eight: 2MHz/8 = 250 KHz.

To arrive at 2400 Hz and 1850 Hz, the signal must be divided by 104 and 135 at ICs J and K. Since the counters can only divide by 256, they can be started at a count other than 0. They can be wired to load the flip-flop with any starting count when they get a load pulse. Since their highest count is 255, 104 and 135 are subtracted from 255, which leaves 151 and 120 for the start count. Then, depending on whether they get a logic "1" or "0," they will give the proper division of the 2MHz signal.

The carry output is sent to IC E and divided by eight. Output data is shown on the right side of the schematic. A TTL modulated square wave is at H8. A recorder cannot accept this because it wants to see a MIC level signal in the audio range. A square wave contains harmonics in the megahertz band. R50 and R51 act as a voltage divider to reduce the output to a few hundred millivolts. R41 and C22 integrate the signal into a sawtooth. It is then output to the tape deck MIC input. (Note that R50 should be 22K rather than 220K.)

One Slot!



Almost too good to be true, the Altair 16K Static RAM board is easily the most advanced memory module yet developed for the Altair 8800, 8800a and 8800b computers.

Four Altair 16K Static boards add up to the entire 64K of memory directly accessible by the Altair.

The Altair 16K Static board offers two surprise features—minimal power requirements and fast access time. One Altair 16K Static board draws less current than any 8800 compatible 4K boards, thus four Altair 16K Static boards can be plugged into the Altair 8800 without beefing up the power supply.

The maximum access time of the Altair 16K Static board is 215 nanoseconds, which makes this board **the fastest Altair compatible static board in existence.**

The Altair 16K Static is now in full production. Special introductory price is $765 in kit form and $945 assembled.

MAIL THIS COUPON TODAY

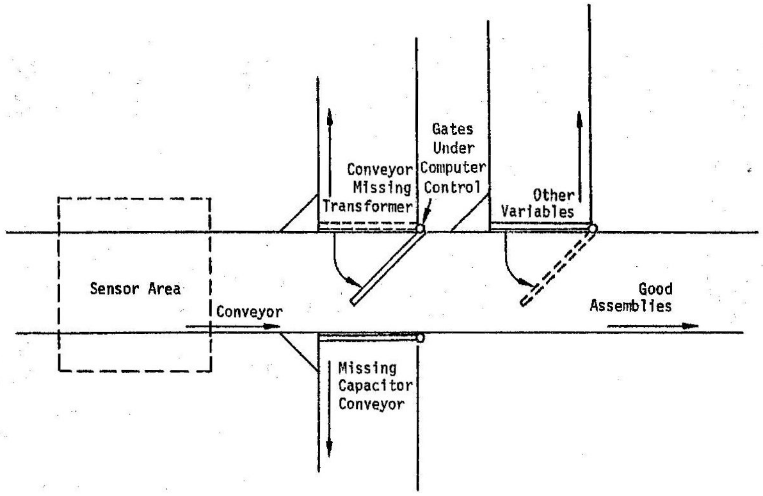
Enclosed is check for $__________
BankAmericard #__________
or Master Charge #__________
☐ Altair 16K Static ☐ Kit ☐ Assembled
(include $3 for postage and handling)
☐ Please send free information package and price sheet.
NAME__________
ADDRESS__________
CITY__________ STATE AND ZIP__________
MITS/2450 Alamo SE/Albuquerque, NM 87106/505-243-7821

Prices, delivery and specifications subject to change. Allow up to 60 days for delivery.

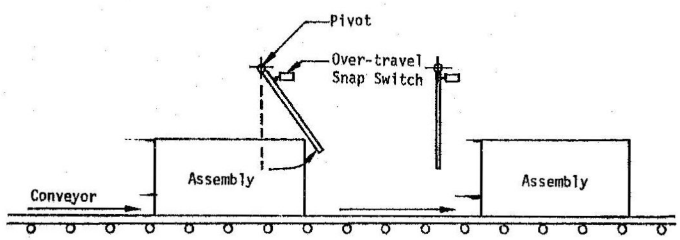
2450 Alamo SE/Albuquerque, NM 87106/505-243-7821

If the conveyor speed is not constant, a tachometer can be attached to the conveyor and its signal used to clock in the sensor array data.

If the computer does detect variations from the stored profile, it can be programmed to initiate action through the PCI board, based on what variations are sensed. For example, the conveyor might have movable gates, which are activated by the relay outputs of the PCI board, at either side. The gates can be positioned further down the conveyor path.

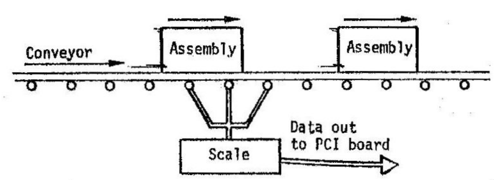


When these gates are extended across the conveyor, the assembly is deflected off the main conveyor and onto another, carrying it to the re-work sections of the plant, salvage or whatever. The computer can determine when the correct amount of time has passed so that the rejected assembly is near the proper gate before it activates the gate. The relay outputs can switch a solenoid to drive the gate directly or operate a solenoid valve to operate the gate hydraulically or pneumatically. The output can also operate a press that stamps the good assemblies with a "pass" stamp. The computer can keep track of the number of units passing it, the number rejected and the reasons for rejection, thus generating a continuous production report.



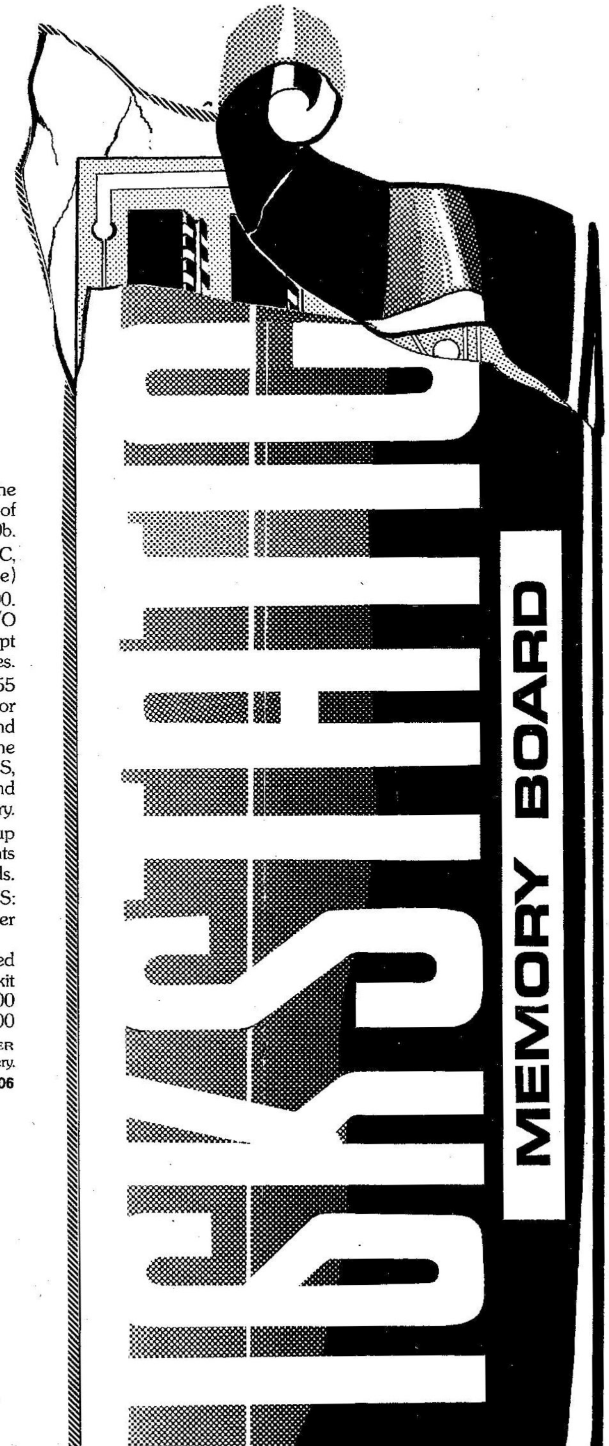
Other sources of data for inspection are mechanical "finger" switches mounted above and to the sides of the conveyor. The switches can be different lengths to sense different sizes of objects. The fingers can also actuate multiple switches to detect different amounts of deflection.

For installation of a heavy part, a closed assembly might be checked internally where optical or dimensional sensors cannot "see," by installing a scale with digital output under the conveyor system.



This is only one possible application of a microcomputer in an industrial environment. In principle, microcomputers can be adapted to almost any repetitive chore in an industrial plant, leaving the human workers free for more interesting, creative and valuable jobs.

Something Sweet for your altair T.M. 680-b



MITS is pleased to announce the development of a 16K static card for the Altair 680b. With an access time of 215 nanoseconds and low power consumption of 5 watts, we feel that this is an excellent addition to the Altair 680b.

To sweeten the pot even more, we are including a free copy of Altair 680 BASIC, assembler, and text editor on paper tape. ($275 value)

Altair 680 BASIC is identical to the 8K BASIC developed for the Altair 8800. Features include Boolean operators, the ability to read or write a byte from any I/O port or memory location, multiple statements per line, and the ability to interrupt program execution and then continue after the examination of variable values.

Other features of Altair 680 BASIC include variable length strings (up to 255 characters), with LEFT$, RIGHT$ and MID$ functions, a concatenation operator and VAL and STR$ to convert between strings and numbers. Both string and numeric arrays of up to 30 dimensions can be used. Nesting of loops and subroutine calls is limited only by available memory. Intrinsic functions include: SIN, COS, TAN, LOG, EXP, SQR, SGN, ABS, INT, FRE, RND and POS, in addition to TAB and SPC in PRINT statements. Altair 680 BASIC takes 7K bytes of memory.

MITS has also developed an expander card for the Altair 680b that lets you add up to three boards inside the main case. Read "Computer Notes" for announcements of additional Altair 680b boards.

PRICES:

Altair 680-BSM, 16K Static Memory Board, including Altair 680 BASIC, assembler and text editor . . . . $685.00 kit
$865.00 assembled

Altair 680-MB Expander Card with one Edge Connector . . . . $24.00 kit

Altair 680 BASIC (purchased separately) $200.00

Altair 680 assembler and text editor (purchased separately) . . . . $ 75.00

PRICE APPLIES ONLY TO PURCHASERS OF ALTAIR 680b COMPUTER

Prices, specifications subject to change. Allow 30-60 days for delivery.

MITS, Inc. 2450 Alamo S.E./Albuquerque, New Mexico 87106

wacc II

May 4-7

World Altair Computer Convention

Convention Center
Albuquerque, New Mexico